

A Cognitively Relevant Lexical Semantics

MSc Dissertation

Cornelia M. Verspoor
`kversp@cogsci.ed.ac.uk`
Centre for Cognitive Science
University of Edinburgh

21 September 1994

Abstract

This project is an attempt to capture verbal syntactic alternations which stem from underlying semantic differences by means of a psycholinguistically motivated lexical representation. It involves development of a representation for capturing the structure of syntactically relevant semantics and specification of lexical rules which determine whether particular verbs can alternate. The representation will be based on the work of Steven Pinker (1989) and Ray Jackendoff (1990) and integrated into the framework of Head Driven Phrase Structure Grammar (HPSG).

Acknowledgements

It would be impossible to undertake a project of this size without intellectual, monetary, and emotional support.

My intellectual support came from my supervisor, Alexander Holt, who allowed me to run with a vague idea and an even vaguer project proposal. I thank him for having faith in my ability to produce something worthwhile. Tomaz Erjavec, Claire Grover and Carl Vogel also deserve thanks, Tomaz for putting up with my naive ALE questions, Claire for her HPSG wisdom, and Carl for getting me past dead-ends and back on track to a solution. In addition, I would like to acknowledge the high level of intellectual curiosity at the Centre for Cognitive Science, without which I couldn't have maintained motivation for the duration of this work.

The support of Betty Hughes and the Computing Support Team must also be acknowledged. Without someone to keep track of all the chaos around here, or without a quick response to the system crashes, noone would be able to get anything done.

My monetary support came in the form of an education subsidy from the International Bank for Reconstruction and Development, my father's employers. I would like to thank them for giving me the sustenance that I needed to spend the year here. I hope that this will give them some encouragement to continue supporting the higher education of their employees' children.

Last and certainly not least, I would like to thank my family and friends for helping me maintain sanity. Mom and Dad may have been thousands of miles away, with hundreds of things on their mind, but they still took time every weekend to phone me and make sure I wasn't working too hard! They also provided me with plane tickets to escape Edinburgh from time to time. I want to thank Matt Davis for innumerable heart-to-hearts, drinks, and meals, Zelal Gungordu for smiling and listening, and Bernie Jones for putting up with my eccentricities and reading a draft of this thesis without laughing.

Contents

Abstract	i
Acknowledgements	ii
Introduction	1
1 Representation	2
1.1 Background	2
1.1.1 Jackendoff's conceptual structures	2
1.1.2 Pinker's lexical semantic structures	7
1.2 The integration of the structures	12
1.2.1 Conceptual Constituents	13
1.2.2 Action and Thematic Tiers	13
1.2.3 Semantic Description	14
1.2.4 STATE and EVENT Predicates	15
1.2.5 A Word on Notation	19
1.2.6 Subordinating Functions	21
1.2.7 Manner	23
1.2.8 Time	23
1.2.9 Properties	24
1.2.10 Semantic Field specification	30
1.2.11 Summary of Differences	32
2 The Dative Alternation	36
2.1 Description	36
2.2 Jackendoff's Explanation	37
2.3 Pinker's Explanation	39
2.3.1 Subclass Definitions	40
2.3.2 Lexical rules	41
2.4 The Lexicon	42
2.5 Lexical Rules	43
2.5.1 "to" Datives	44
2.5.2 "for" Datives	54

3	Discussion	59
3.1	Lexical Rules	59
3.1.1	The Number of Lexical Rules	59
3.1.2	Violations of Lexical Rules	60
3.2	The Dative Alternation	61
3.2.1	The Morphophonological Constraint on the Dative	61
3.2.2	Directionality	62
3.3	Implementational Issues	63
3.3.1	Arguments vs. Adjuncts	63
3.3.2	Selectional Restrictions and the Double Object Form	64
3.3.3	Lexical Issues	65
3.3.4	An Enriched Type Hierarchy vs. Enriched Semantics	65
3.3.5	Unification and Defaults	66
3.4	Semantic Issues	67
3.4.1	Compositionality	67
3.4.2	Inferencing	67
3.4.3	Quantification	68
4	Conclusions	69
4.1	Jackendoff vs. Pinker	69
4.2	Reduction of Lexical Redundancy	70
4.3	Psychological Validity	70
4.4	Topics for Further Research	71
A	Glossary	72
B	Representational Inventories	73
B.1	Jackendoff	73
B.2	Pinker	74
B.3	Verspoor	75
C	Lexical Entries	76
C.1	Specified Lexical entries	76
C.2	Generated Lexical entries	85
D	ALE Files	92
D.1	Type Hierarchy	92
D.2	Lexical Rules	98
D.3	Grammar Rules	110
	Bibliography	122

Introduction

Semantic representation in much existing computational work has largely occurred within theoretical frameworks which are not focused on addressing semantic issues at the lexical level. The focus on the compositionality of words, characterised in terms of lexical entries, and sentence-level semantic integration has overshadowed any discussion of the form of the lexical entries themselves. The result is that semantic generalisations derived from features of word meaning which interact with syntax have largely been ignored. Such interactions provide a method of reducing lexical redundancy in natural language processing systems.

The representation utilised in this thesis is developed from semantic regularities identified through both child language acquisition studies and linguistic analysis, presented in the work of Stephen Pinker (1989) and Ray Jackendoff (1990). Their proposals for semantic representation will be introduced and compared. A representation combining insights from each of these proposals will be outlined and shown to effectively capture syntactically relevant semantic information.

There are several syntactic phenomena which can be explained in terms of the underlying semantics of the words which participate in the phenomena. The particular grammatical phenomenon to which the implemented representation will be applied is that of the dative alternation, although the approach is easily generalisable to other phenomena such as the causative and locative alternations. Through this application it will be shown that differences in syntactic form have particular semantic consequences which can be captured succinctly in the form of lexical rules. These lexical rules are applied to verb lexical entries, enabling a significant reduction in lexicon size.

Semantic structures for verbs built from the developed semantic representation will be integrated into the framework of Head-driven Phrase Structure Grammar (HPSG). HPSG is a unification-based linguistic theory, utilising lexical entries to provide a direct interface between syntax and semantics. The mechanisms of the theory supply the mapping from semantic structure to syntactic form, thus enabling a focus on the development of an appropriate semantic structure for handling semantic regularities. The current document will not include any detailed discussion of HPSG and will assume a basic familiarity with its constructs and mechanisms. The reader is referred to Pollard and Sag 1994 for an explanation of the theory.

The outline of this document is as follows. Chapter 1 focuses on the semantic representation, reviewing the work of Jackendoff and Pinker and describing how elements of each were integrated for the representation used in the implementation. Chapter 2 discusses the dative alternation, overviewing the Jackendoff and Pinker explanations, and showing how Pinker's explanation was adapted in the implementation accompanying this thesis to adequately model the dative alternation and its semantic effects. Chapter 3 discusses several issues that came up in the course of developing and implementing the model of the dative alternation.

Chapter 1

Representation

The types of syntactic phenomena to be addressed in this thesis centre on alternations in verb argument structure. It is the syntactic requirements of a verb in a sentence which are critical for determining the grammaticality of the sentence. These syntactic requirements are often dependent on elements of the verb’s semantics. As a result, the representations discussed throughout this thesis will be of verb semantics.

The semantics of other elements in the sentence, such as the noun phrases, are certainly relevant to sentence grammaticality, but are not central to the account of the syntactic alternations. Pinker does not discuss the representation of any words which are not verbs, and Jackendoff relies mainly on a 3D physical representation, which is not decomposed into semantic primitives or discussed in terms of its internal structure, to capture the semantics of objects. The only discussion of semantics outside of verb representation in this thesis will have to do with properties associated with nouns, but no formal representation for the nouns or their associated properties will be proposed.

1.1 Background

1.1.1 Jackendoff’s conceptual structures

Ray Jackendoff (1990) proposes a theory of *conceptual structures* which are characterisations of the conceptual knowledge represented in the human mind. Such mental structures form the meanings of linguistic expressions. Jackendoff’s goal is to represent the form of “I-language”¹, or the internal language. I-language is distinct from “E-language”, which is language viewed as an external artifact which existing independently of any users. He further wishes to capture the formal relations between the level of conceptual structures (I-language representation) and other levels of representation, such as that of syntax.

Conceptual concepts are built up from lexical concepts which capture the conceptual information associated with words. Lexical concepts in turn are represented as structural schemas built from primitives, which can be compared for compatibility with the mental representations of arbitrary new objects. *Conceptual formation rules* specify combination of the conceptual primitives, defining permissible structural relations and accounting for the apparently infinite human concept generation capacity through their recursive nature.

¹Jackendoff borrows the terms *I-language* and *E-language* from Chomsky (1986).

Jackendoff is interested in identifying the formal relations that exist between different uses of a lexical item and whether such relations have general applicability. Examples of the type of data which he is interested in accounting for appear in (1.1) and (1.2)². The sentences in (1.1) suggest related but differing uses of the same verb, while there does not appear to be any difference in the meaning expressed by the verbs in (1.2).

- (1.1) a. The window broke.
 Bill broke the window.
- b. Harry climbed the mountain.
 Harry climbed down the mountain.
- c. Max looked at the insects.
 Max looked for the insects.
 Max looked smart.
- (1.2) a. Harry bought a yoyo in Chicago.
 Harry bought a yoyo.
- b. The clock ran down on Tuesday.
 The clock ran down.

Jackendoff views conceptual structure as a “necessary conduit” (1990, p. 34) to syntactic structure, and hence wishes mainly to represent grammatically-relevant semantic distinctions. He therefore proposes a small set of conceptual primitives, or *conceptual constituents*, corresponding to ontological categories such as EVENT, STATE, THING, PATH and PLACE with which he hopes to characterise most grammatically relevant semantic distinctions. He does not decompose all semantics into features, relying instead upon words and the association of an unelaborated physical object representation (3D model) with lexical entries to capture many semantic properties.

The semantics of conceptual structure is construed as parallelling Chomsky’s X-bar syntax. This parallelism exists in the sense that the conceptual constituents share important properties which can be formally summarised, just as the major syntactic categories share properties. Such properties are best expressed in the formation rules for X-bar syntax and “X-bar semantics”. The rules for X-bar syntax are found in (1.3), and the general schema for X-bar semantics rules appears in (1.4).

- (1.3) $XP \rightarrow \text{Spec} - X'$
 $X' \rightarrow X - \text{Comp}$
 $X \rightarrow [\pm N, \pm V]$

- (1.4) $[\text{Entity}] \rightarrow \left[\begin{array}{l} \text{ConceptualConstituent} \\ \text{Token/Type} \\ F(\text{Entity}_1, \text{Entity}_2, \dots) \end{array} \right]$

Here, F ranges over functions which can expand conceptual constituents, and the *Entity* elements are the arguments to the function. The *Entity* elements can themselves be expanded by the formation rule (1.4), and thus the rule provides recursion. Functions are n -ary ($n \geq 0$) which take other Entities as arguments. 0-ary functions are unelaborated concepts,

²From Jackendoff 1990, pp. 20-21.

represented by words. The labelling of the Entity as a *Token* or a *Type* depends on whether it refers to an object in the world (a *token*) or to a class or category (a *type*).

In the attempt to capture semantic structure across domains, there is a trade-off between addition of functional primitives to the system and elaboration of existing primitives through a feature system. Jackendoff does not propose any systematic way of deciding among the two; he suggests that for the moment each case must be analysed in turn for the most effective representational solution when the existing set of primitives is inadequate. The set of functions currently in use in Jackendoff’s theory is outlined below. “Event-functions” are those functions which elaborate an EVENT, and “state-functions” are those functions which elaborate a STATE.

- GO an Event-function which denotes motion along a path; two arguments: the Thing in motion and the Path it traverses.
- STAY an Event-function which denotes stasis over a period of time; two arguments: the Thing standing still and its location.
- BE a State-function for specifying the location of objects.
- ORIENT a State-function for specifying the orientation of objects.
- EXT a State-function for the spatial extension of linear objects along a path.
- CONF a State-function that expresses that a verb describes the internal spatial configuration of an object; one argument: the Thing that is in the spatial configuration.
- CAUSE an Event-function specifying cause and effect relations; two arguments: a Thing which is the agent or an Event which is the cause, and an Event which is the effect.
- INCH an Event-function specifying the “inchoative”, a change taking place which has a final state; one argument: the State in which the event terminates.
- MOVE an Event-function which specifies that an object is moving/moves; one argument: the Thing which is moving.
- EXCH a modifying function specifying an event which is “in exchange for” the entity being modified.
- AFF a formal elaboration of an Event which specifies that an actor “affects” a patient; two arguments: the actor and the patient.
- REACT a formal elaboration of an Event which specifies that one Thing reacts to another; two arguments: X, Y; X “reacts to” Y.
- place functions at, on, in, under, ... functions expressing location.
- path functions to, from, toward, away-from, via; functions expressing direction.

These functions may be augmented in several ways in the representation. First, the functions may be extended to fields other than that in which they originated. By doing this, Jackendoff adopts Gruber’s (1965) Thematic Relations Hypothesis, which accounts for the fact that verbs and prepositions can appear in multiple semantic fields via systematic paradigms by claiming that they are each realisations of the basic conceptual functions. For example, the GO function may not only express physical motion but also a change of possession. The paradigms are distinguished from one another by a semantic field feature, which specifies how the components of the structure are to be interpreted. This feature is specified with a subscript; e.g. GO_{POSS}. Second, certain semantic differences between expressions are not great enough to warrant postulation of new functions. Rather, the differences are better captured through the use of a feature which reflects the semantic nuances of the function. As an example, consider the difference in (1.5)³.

³Examples (1.5), (1.6) and (1.7) are from Jackendoff 1990, p. 108.

- (1.5) a. The cockroach ran into the wall.
b. Bill ran into the wall.

In (1.5a), the cockroach does not make contact with the wall, whereas in (1.5b), Bill does make contact with the wall, despite the parallel structures and word choice. The distinction can be formalised as in (1.6).

- (1.6) a. *Noncontact* “into”
[_{PATH} TO([_{PLACE} IN-**contact**([])))]
b. *Contact* “into”
[_{PATH} TO([_{PLACE} AT+**contact**([])))]

Verbs which explicitly involve contact, such as verbs of touching, can also incorporate this feature, as in (1.7). Features therefore enable specification of semantic properties which are enrichments of more general semantic relations.

- (1.7) NP_i touch NP_j
[_{STATE} BE+**contact**([]_i, [_{PLACE} AT+**contact**([]_j)))]

Conceptual constituents can be elaborated in terms of functions and their arguments, in the form of the general schema in (1.4). Examples of such formation rules are found below. Those listed are immediately relevant to representation of verb semantics. Other decompositions of conceptual constituents into function-argument structure, such as those pertaining to THING or PROPERTY are not shown⁴.

- [THING] → [_{THING} JOHN/MARY/BOOK/...]
- [PLACE] → { [_{PLACE} place function ([_{PROPERTY}])]
[_{PLACE} place function ([_{THING}])] }
- [PATH] → [_{PATH} { $\begin{matrix} to \\ from \\ toward \\ away-from \\ via \end{matrix}$ } ([[_{THING}]])]
- [EVENT] → { [_{EVENT} GO ([_{THING}] , [_{PATH}])]
[_{EVENT} STAY ([_{THING}] , [_{PLACE}])]
[_{EVENT} CAUSE ([[_{THING}]] , [_{EVENT}])]
[_{EVENT} INCH ([_{STATE}])]
[_{EVENT} MOVE ([_{THING}])] }

⁴The decomposition of a THING into function-argument structure is used for elaborating nouns, e.g. in *father of the bride*, one THING (*father*) has another THING as an argument (*bride*), and the decomposition of a PROPERTY is used for elaborating properties, e.g. in *afraid of Harry*, the PROPERTY (*afraid*) has a THING argument (*Harry*).

$$\bullet \text{ [STATE]} \rightarrow \left\{ \begin{array}{l} \left[\begin{array}{cc} \text{STATE} & \text{CONF} \end{array} \left(\left[\begin{array}{c} \text{THING} \end{array} \right] \right) \right] \\ \left[\begin{array}{cc} \text{STATE} & \text{BE} \end{array} \left(\left[\begin{array}{c} \text{THING} \end{array} \right], \left[\begin{array}{c} \text{PLACE} \end{array} \right] \right) \right] \\ \left[\begin{array}{cc} \text{STATE} & \text{ORIENT} \end{array} \left(\left[\begin{array}{c} \text{THING} \end{array} \right], \left[\begin{array}{c} \text{PATH} \end{array} \right] \right) \right] \\ \left[\begin{array}{cc} \text{STATE} & \text{EXT} \end{array} \left(\left[\begin{array}{c} \text{THING} \end{array} \right], \left[\begin{array}{c} \text{PATH} \end{array} \right] \right) \right] \end{array} \right\}$$

The above examples fall under what Jackendoff refers to as the *thematic tier*, which is the set of conceptual functions relating to motion and location. To clarify actor/patient relations in the semantics of words, he proposes an *action tier*. These relations can informally be characterised in terms of the frame in (1.8), where X is the Actor and Y is the Patient.

$$(1.8) \quad \left\{ \begin{array}{l} \text{What happened} \\ \text{What X did} \end{array} \right\} \text{ to Y was...}$$

The tier is formally specified with the addition of a representation of the functions AFF (“affect”) and REACT and their arguments to the representations of the thematic tier. The schema for this addition is found in (1.9) (Jackendoff 1990, p. 127). The “...” represents the part of a formation rule which already exists, which is simply augmented with the action tier information.

$$(1.9) \quad \text{[EVENT]} \rightarrow \left[\begin{array}{cc} \dots & \\ \text{EVENT} & \text{AFF} \end{array} \left(\left\langle \left[\begin{array}{c} \text{THING} \end{array} \right] \right\rangle, \left\langle \left[\begin{array}{c} \text{THING} \end{array} \right] \right\rangle \right) \right]$$

The two arguments of AFF are the actor and the patient. Since the semantics of a verb need not specify both of these roles, Jackendoff adopts the representational conventions in (1.10) (Jackendoff 1990, p. 128).

$$(1.10) \quad \begin{array}{ll} \text{a. } [\text{AFF}(\text{[X]}, \text{ })] & (\text{X} = \text{Actor only}) \\ \text{b. } [\text{AFF}(\text{ }, [\text{Y}])] & (\text{Y} = \text{Patient only}) \\ \text{c. } [\text{AFF}(\text{[]}, [\text{Y}])] & (\text{implicit Actor}) \\ \text{d. } [\text{AFF}(\text{[X]}, [\text{]})] & (\text{implicit Patient}) \end{array}$$

The alternative action function is REACT which, like AFF, takes two arguments. REACT([X],[Y]) is understood as “X reacts to Y”. Both of these functions may be elaborated with features. This featural elaboration is particularly useful in capturing the difference between having a positive, negative, or neutral effect on the second argument (AFF⁺, AFF⁰, and AFF[−] respectively). A positive effect corresponds to helping, a negative effect to hindering, and a neutral effect corresponds to allowing the second argument to do something.

Adjuncts can be viewed as conceptual constituents which modify other constituents. Jackendoff introduces a *restrictive modification schema* for the incorporation of modifying information into a conceptual structure. It can be seen in (1.11). An example of the use of this schema is found in (1.12) (Jackendoff 1990, p. 56).

$$(1.11) \quad \text{[Entity}_1\text{]} \rightarrow \left[\begin{array}{c} \text{Entity}_1 \\ \text{[Entity}_2\text{]} \end{array} \right]$$

$$(1.12) \quad \text{red hat} \\ \left[\begin{array}{cc} & \text{HAT} \\ \text{THING} & \left[\begin{array}{c} \text{PROPERTY RED} \end{array} \right] \end{array} \right]$$

It is possible that the modifying information is a STATE or EVENT which modifies another STATE or EVENT. Jackendoff introduces several *subordinating functions* which capture the relationship between the modifier (X) and the modified (Y). These are generally used to incorporate information added by an adjunct into a conceptual structure. This incorporation is accomplished through *correspondence rules* indicating the syntactic structures which correspond to particular conceptual relations.

- BY X is the means to accomplish Y.
- FROM X is the reason why Y occurs.
- FOR X is the intended result of the action in Y.
- WITH The event/state X accompanies the event/state Y.
- EXCH Y occurs in exchange for X.

Jackendoff justifies his choices of conceptual constituents, functions, function features, and formation rules by showing how they can account for subtle differences in meaning. Furthermore, he shows how the lexical conceptual structures interact with one another and with syntax to provide interpretations of complete sentences. This interaction occurs via *linking rules* and *adjunct rules* which explicitly specify mappings between structural relations and syntactic positions. The linking theory is designed to provide a principled explanation of the syntax/semantics mapping. Without the linking theory, Jackendoff would rely on coindexing of syntactic subcategorised arguments and semantic conceptual argument positions to establish the mapping between syntax and semantics. However, complete stipulation of the correspondence of syntactic arguments and semantic arguments in the lexical entry does not adequately constrain the possible relationships between the arguments. The theory which addresses this issue will not, however, be fully laid out here, as it will not be adopted in this work.

1.1.2 Pinker's lexical semantic structures

The focus of Steven Pinker's (1989) work is on outlining a theory of the form of linguistic knowledge which also accounts for child acquisition of argument structure. His goal is to develop linguistic explanations for variations in verbal argument structure, as exemplified in the contrasts in (1.13) and (1.14), which are supported not only by adult linguistic data but also by child language acquisition data.

- (1.13) a. John taught the song to Bill.
 b. John taught Bill the song.

- (1.14) a. John yodeled the song to Bill.
 b. *John yodeled Bill the song.

Pinker's analysis of the adult data involves identification of verb classes which are semantically related, but which appear in contrasting syntactic configurations. This method highlights features of verb semantics which are critical for syntax while abstracting away from semantic differences which appear to have no syntactic relevance. It attempts to capture the differences between verbs such as *cut*, *break*, *hit*, and *touch* which cause them to behave differently syntactically, despite their apparent cognitive similarity. Pinker thus wishes to distinguish between semantic elements which have relevance for cognitive similarity, such as

the characteristic features of a verb's meaning, and the aspects of that meaning which are consistently linguistically relevant.

The basis of Pinker's semantic representation, then, is a set of semantic elements and relations that is much smaller than the cognitively available and culturally salient distinctions around which verb meanings are organised. Pinker calls this set the *Grammatically Relevant Subsystem*. He emphasises, however, that this set is not aimed at capturing the full meaning of a verb. As described by Pinker (1989, p. 168), "the verb definitions sought will be hybrid structures, consisting of a scaffolding of universal, recurring, grammatically relevant meaning elements plus slots for bits of conceptual information". That unelaborated conceptual information is notated with quotation marks in his representation.

For selection of the semantic elements recurrent in verb meaning, Pinker relies mainly on analysis performed by Talmy (1985). The elements which Talmy identified as prevalent are listed below (adapted from Pinker 1989, pp. 174-175)

1. **"main event"**: a position, state or motion predicated of a theme; the backbone of the verb's meaning.
2. **path, direction, location**
3. **causation**: whether an event has been caused or simply occurs.
4. **manner**: how an actor acts or a theme changes.
5. **properties**: certain properties of a theme or actor with respect to a predicate.
6. **aspect and phase**
7. **purpose or goal**
8. **coreferentiality**: verbs may encode speaker-reference; e.g. *to comb* means to comb someone else's hair in some languages, while it means to comb one's own hair in others.
9. **truth value**: propositional attitudes, speaker assumptions.

Pinker's representational choices are largely adopted from Jackendoff's work, with minor modifications to display more accurately the semantic relations needed to capture the syntactic alternations of the dative, the causative, the locative, and the passive in lexical rules. For an inventory of the representation, see Appendix B.2.

Conceptual constituents are names for the basic conceptual categories, among which all concepts can be divided. The conceptual constituents Pinker utilises are EVENT, STATE, THING, PATH, PLACE, PROPERTY, MANNER. They can be combined into more complex concepts through a set of formation rules. Each consists of an n -ary function ($n \geq 0$) and n arguments. A function may have optional arguments; this is simply a short form for collapsing related functions into one rule. The functions which Pinker adopts include **place-functions**, which specify a place or region with respect to some object, and **path-functions**, which specify a direction towards an object or region. English prepositions are used as mnemonics for these functions. Other permissible functions for combining conceptual constituents are built from two features: \pm dynamic and \pm control. The dynamic feature picks out whether the unmarked kind of constituent type in which the predicate is found is an EVENT or STATE. The control feature picks out whether in the unmarked case the first argument must be animate and in control of the event/state. The two features together define four basic functions, as shown in (1.15) (from Pinker 1989, p. 195).

(1.15)

<i>Predicates</i>	<i>Features</i>	
	Dynamic	Control
GO	+	−
BE	−	−
HAVE	−	+
ACT	+	+

Both EVENTS and STATES can be annotated with a semantic field to indicate a nonspatial field to which the interpretation of the function has been extended. Pinker essentially follows Jackendoff with this type of annotation, adopting Gruber's (1965) Thematic Relations Hypothesis, although they differ on which particular semantic fields are defined. See Section 1.2.10 for more detail on semantic field definition.

Examples of the formation rules Pinker introduces are found below⁵. The angle brackets $\langle \rangle$ around a constituent indicate that the constituent is optional. The curly braces $\{ \}$ represent a set from which one element is chosen for each instance of a formation rule in the specified form. The rule for PROPERTY is an attempt to represent the grammatically relevant properties which might be specified in a semantic structure. Other properties are opaque to lexical rules. Properties can be attached to EVENTS, THINGS, etc.

$$\bullet \text{ PROPERTY} \rightarrow \left\{ \left(\begin{array}{l} \text{animate} \left(\begin{array}{l} \text{human} \\ \text{nonhuman} \end{array} \right) \\ \text{inanimate} \left(\begin{array}{l} 0D \\ 1D \\ 2D \\ 3D \end{array} \right) \\ \text{count} \left(\begin{array}{l} \text{rigid} \\ \text{flexible} \end{array} \right) \\ \text{mass} \left(\begin{array}{l} \text{substance} \left(\begin{array}{l} \text{liquid} \\ \text{semisolid} \end{array} \right) \\ \text{Parts} \\ | \\ \text{PROPERTY} \end{array} \right) \\ \text{aggregate} \rightarrow \end{array} \right) \right\} \\
 \text{(unelaborated property-type concept)}$$

$$\bullet \text{ MANNER} \rightarrow \text{(unelaborated manner-type concept)}$$

$$\bullet \text{ PLACE} \rightarrow \left\{ \begin{array}{l} \text{in} \\ \text{on} \\ \text{under} \\ \text{around} \\ \vdots \end{array} \right\} \left(\text{THING} \right)$$

⁵I utilise Jackendoff's linear notation here for space efficiency. Pinker prefers a tree-structure notation which corresponds directly to the function application form.

- $\text{PATH} \rightarrow \left\{ \begin{array}{c} \textit{from} \\ \textit{to} \\ \textit{away-from} \\ \textit{toward} \\ \textit{via} \end{array} \right\} \left(\left\{ \begin{array}{c} \text{THING} \\ \text{PLACE} \end{array} \right\} \right)$
- $\text{EVENT} \rightarrow \left\{ \begin{array}{l} \text{GO (THING, \langle PATH \rangle, \langle MANNER \rangle)} \\ \text{GO (THING, PROPERTY)} \\ \text{ACT (THING, \langle MANNER \rangle)} \\ \text{ACT (THING, THING, \langle MANNER \rangle)} \end{array} \right\}$
- $\text{STATE} \rightarrow \left\{ \begin{array}{l} \text{BE (THING, PLACE)} \\ \text{HAVE (THING, THING)} \end{array} \right\}$

There are no restrictions on which conceptual constituents can appear as arguments for the various structures; the actual constituents which appear in the formation rules as valid arguments for particular functions are determined by the semantics of existing English verbs which have been analysed. It would in theory be possible for a verb to exist for which its semantics involves ACTing upon a PLACE. This would simply require the addition of a formation rule to accommodate the new relation. On the other hand, since the arguments in formation rules must be composed of conceptual constituents, the formalism does predict that no verb can have a meaning in which multiple events are conjoined together without any subordinating relation between them. Thus, to use Pinker's example, there are no verbs meaning "Simultaneously, John yawned and the cat fell off the roof" (Pinker 1989, p. 196).

The structures generated with the formation rules for EVENT or STATE can be augmented with semantic structures through the use of subordinating relations. Thus, any rule of the form in (1.16a) can be augmented to the form in (1.16b), where *SubordFunc* stands for the characterisation of a particular subordinating relation.

- (1.16) a. $\left\{ \begin{array}{c} \text{EVENT} \\ \text{STATE} \end{array} \right\} \rightarrow \text{Function} \left(\textit{arg}_1, \textit{arg}_2, \dots \right)$
- b. $\left\{ \begin{array}{c} \text{EVENT} \\ \text{STATE} \end{array} \right\} \rightarrow \text{Function} \left(\textit{arg}_1, \textit{arg}_2, \dots, \text{SubordFunc} \left(\left\{ \begin{array}{c} \text{EVENT} \\ \text{STATE} \end{array} \right\} \right) \right)$

Subordinating relations therefore allow more precise specification of verb semantics. Like the functions expanding conceptual constituents, the subordinating relations can be characterised in terms of a set of features, this time a sextuple of features.

- \pm **focus** +focus corresponds to a focus on the causing action; –focus corresponds to a focus on the effect.
- \pm **potency** potency refers to whether or not the agent of the subordinated event succeeds in exerting the change indicated in the event on the patient. +potency indicates that he succeeds; –potency indicates that he fails.
- \pm **cause-occurrence** the cause event occurs or fails to occur.
- \pm **effect-occurrence** the effect event occurs or fails to occur.

- **±purposive** +purposive indicates that the final effect is the purpose or goal of the agent; –purposive indicates that the agent does not intend the final effect.
- **±deontic** +deontic indicates that an event incurs an obligation or fulfills one; –deontic indicates that obligation is not part of the meaning of the verb.

Pinker again opts to use mnemonics to clarify the interaction between the above features. Some of the subordinating relations found in the semantics of English verbs are summarised in (1.17)⁶.

(1.17)

<i>Subordinating Relation</i>	<i>Features</i>					
	focus	potency	cause-occ.	effect-occ.	purposive	deontic
effect	+	+	+	+	–	–
cause	–	+	+	+	–	–
despite	–	–	–	–	–	–
but	+	–	–	–	–	–
let	+	+	+	+	–	–
prevent	+	+	+	–	–	–
means	–	+	+	+	+	–
for/to	+	+	+	+	+	–
obligates	–	+	+	+	–	+
fulfills	+	+	+	+	–	+

The final representational mechanisms which Pinker introduces are aimed at handling temporal elements of verb meaning and relations between within-verb subevents. The main temporal representation is done on a time-line. States are regions on the line with no distinct boundaries; instantaneous events are points; accomplishments are regions bounded at their ends by a point; achievements are points bounding the end of a region. Each EVENT and STATE in the verbal semantics is mapped to some part of the line.

Semantic constraints on the syntactic alternations are captured in *lexical rules*. The lexical rules are defined in terms of semantic structures, specifying a mapping from one particular semantic structure (the input form) to another (the output form). Surface syntax derives from semantic structures through *linking rules* indicating correspondences between structural argument positions and syntactic argument positions. Structural argument positions which can participate in linking rules are known as *open arguments*, and are marked in the semantic representation⁷. A lexical rule is blocked from applying to verbs whose semantic structure is incompatible with the input of the rule. The input requirements must be precisely defined to take into account the function of the lexical rule, so that the rule is blocked from applying to verbs which are cognitively incompatible with the *output* of the rule. For example, verbs which match the input for the dative alternation lexical rule must be required to be capable of denoting prospective possession of the referent of the direct object by the referent of the indirect object, because the output of the lexical rule specifies such a possession⁸.

⁶Adapted from Pinker 1989, p. 202.

⁷Open arguments are indicated in Pinker's tree structural representation by attaching a set of empty brackets [] to the conceptual constituent at the relevant node of the tree. I will not provide an alternative representation in the linear form, as this is purely necessitated by Pinker's approach to syntax, which will not be utilised in this work.

⁸See Section 2.1 for definition of direct and indirect objects

Each type of syntactic alternation can be characterised in terms of a general lexical rule, the *broad-range lexical rule*. The classes of verbs to which each of these general rules apply are known as broad-range conflation classes. The broad-range lexical rule for the dative alternation, for example, transforms a predicate meaning “to cause X to go to Y” into a second predicate meaning “to cause Y to have X”. The linking rules ensure that the initial form corresponds to the prepositional form (as in the (a) sentences in (1.13) and (1.14)), while the second semantic structure maps onto the double object form (as in the associated (b) sentences). This rule accounts for the fact that verbs whose meaning is incompatible with “cause to have” cannot appear in the double object form, as in (1.18).

- (1.18) a. John drove the car to Chicago.
 b. *John drove Chicago the car.

Once such lexical rules have been proposed to explain the syntactic alternations, it remains to show how a particular verb can be construed as having a meaning relevant to the rules, such as “cause to have” in the discussion above. This is accomplished through precise specification of the representation of semantic structures. The broad-range rules are explicitly defined in terms of these representations, and *narrow-range lexical rules* with elaborated semantic structures are added to pick out narrow subclasses of verbs divided according to their syntactic properties. These narrow-range lexical rules seek to outline the semantic differences between alternating and nonalternating verbs that are not captured in the coarse broad-range rules. They are developed through comparison of the semantics of dativisable and nondativisable verbs.

Thus, while broad-range rules provide the *necessary* condition for a verb to alternate, the narrow-range rules provide the *sufficient* conditions by capturing very specific grammatically relevant semantic structures common to the narrow subclasses. The key generalisation about narrow conflation classes is that they make explicit the fact that if a verb alternates, other verbs with the same *grammatically relevant* semantic structure will also alternate.

The development of a semantic representation for a full sentence from the representations of individual words is only weakly specified within Pinker’s framework. He relies on *categorical correspondence rules*, adopted from Jackendoff, which specify that major phrasal categories (NP, VP, ...) correspond to complete conceptual categories. It is also suggested that there are likely universal regularities in the correspondence between phrasal categories and conceptual categories. These rules combined with syntactic specification of argument structure and a parsing mechanism determine how a semantic representation is built up.

In the context of this thesis, Pinker’s evidence in favour of the psychological plausibility of his theory with respect to child acquisition of the structures will not be assessed. The validity of the claim that his theory is cognitively relevant will simply be assumed in the adoption of many of its key concepts.

1.2 The integration of the structures

The theories of Jackendoff and Pinker have much representational mechanics in common. I have integrated them in such a way as to capture the main elements of each, while eliminating any redundancies. A summary of the representation can be found in Appendix B.3. Any structures discussed in this chapter which are represented in either Pinker’s or Jackendoff’s representations will be labelled with a P or a J, respectively, and appear in **Sans Serif** type.

1.2.1 Conceptual Constituents

The basic conceptual categories adopted in the representation for this thesis are EVENT, STATE, THING, PROPERTY, MANNER, PATH, and PLACE. These are the categories which can serve as arguments to the function predicates.

PROPERTY, however, does not appear as an argument in any of the formation rules for the predicates. Properties are, rather, incorporated into semantic representations as semantic conditions on THINGS. Jackendoff and Pinker’s main use of PROPERTY as a function argument is for predication, e.g. *The meat is raw*, represented by Jackendoff as in (1.19) (1990, p. 202). This would be represented in the current system without requiring a different new formation rule for the BE predicate, as in (1.20). Properties will be discussed in detail in Section 1.2.9.

$$(1.19) \quad [{}_{\text{STATE}} \text{BE}_{\text{ident}} (\text{meat}, (\text{at} (\text{raw})))]^J$$

$$(1.20) \quad [{}_{\text{STATE}} \text{BE}_{\text{ident}} (\text{THING}^{\text{meat}}, (\text{at} (\text{THING}^{\text{raw}})))]$$

MANNER is also only used as an argument in a very limited way. It can only be used as an argument of certain predicates. It will be discussed in Section 1.2.7.

1.2.2 Action and Thematic Tiers

The semantic representation for each verb will be captured by a DESCRIPTION. Some verbs will only express a simple STATE or EVENT, while most will incorporate both a thematic tier and an action tier, analogous to Jackendoff’s tier system (see Section 1.1.1). The action tier captures the agentive/patient information through the use of Jackendoff’s AFF function (see the figure in (1.10)). This function takes the place of Pinker’s ACT function. It is important to incorporate the agentive/patient information because of the critical role it plays in capturing the semantics of the dative alternation. As discussed in Section 2.2, Jackendoff identifies the key semantic difference between the dative and double object forms as the Actor/Patient relationship. Similarly, Pinker (see Section 2.3) presents the dative alternation primarily as an alternation of the affected entity in the variant forms. Specifically, the dative form is associated with a structure “X causes Y to go to Z” in which Y is the Patient, while the double object form is associated with “X causes Z to have Y” in which Z is the Patient.

The thematic tier in Jackendoff’s system is aimed at representing conceptual roles having to do with motion and location. This tier is construed to contain only one function and its arguments. Although any STATE or EVENT function may appear in the thematic tier, Pinker (1989, p. 195) argues that this view of the thematic tier cannot provide an adequate representation for the complexity of verb structure, nor does it precisely indicate the types of causal relations that must exist between the two tiers. Pinker chooses to eliminate the distinction between the action and thematic tiers, proposing instead a representation for which the only constraint is that multiple events specified by a verb must stand in some causal relation to one another. These causal relations are expressed via the subordinating functions.

Upon analysis of the structures which Pinker proposes, however, it becomes clear that he does incorporate a structure analogous to the action tier in all complex verb representations. Specifically, each complex verb structure has as its primary event an ACT event, as shown in Pinker’s representation of the verb *break* in (1.21) (Pinker 1989, p. 198).

$$(1.21) \quad \left[\text{ACT} \left(\text{THING}_1, \text{THING}_2, \text{effect} ([{}_{\text{EVENT:identificational}} \text{GO} (\text{THING}_2, \text{PROPERTY}^{\text{“broken”}})) \right) \right]^P$$

All other events are subordinated to a main ACT event. Pinker does acknowledge that “the most obvious subordinating relation is a successful sequence of cause and effect: an action results in some event that is its effect” (1989, p. 197) but does not go so far as to claim that all complex verb semantics involve an ACT event as the most basic relation. For the purpose of this work, it will be assumed that all structures in which multiple occurrences are related must involve an AFF state as the basic occurrence to which all other occurrences are subordinated. This state expresses the agentive/patient relationship of Jackendoff’s action tier, and will thus also be construed as the action tier in this work⁹.

The thematic tier then captures any causal relationships expressed by the verb. This includes the semantic argument structure, i.e. the semantic relationships between the (syntactically) expressed arguments, as well as relations between events. Rather than following Jackendoff in allowing any STATE or EVENT appearing as the occurrence in the thematic tier, this implementation will instead incorporate the observations of Pinker concerning the relationships between events in the semantic structure, and restrict the tier to representing explicitly the relationship between the occurrence captured in the action tier and other STATES or EVENTS through subordinated functions. In particular, the thematic tier will be a *set* of such subordinating functions and their arguments, to accommodate the possibility of a verb indicating multiple relationships among events.

1.2.3 Semantic Description

The grammar for semantic descriptions is as follows:

- $[\text{DESCRIPTION}] \rightarrow \left\{ \begin{array}{c} [\text{STATE}] \\ [\text{EVENT}] \\ [\text{COMPLEX_DESCRIPTION}] \end{array} \right\}$
- $[\text{COMPLEX_DESCRIPTION}] \rightarrow \left\{ \begin{array}{c} [\text{EVENT_DESCRIPTION}] \\ [\text{STATE_DESCRIPTION}] \end{array} \right\}$
- $[\text{EVENT_DESCRIPTION}] \rightarrow \left[\begin{array}{c} \text{EVENT_ACTION_TIER} \\ \text{THEMATIC_TIER} \end{array} \right]$
- $[\text{STATE_DESCRIPTION}] \rightarrow \left[\begin{array}{c} \text{STATE_ACTION_TIER} \\ \text{THEMATIC_TIER} \end{array} \right]$
- $[\text{EVENT_ACTION_TIER}] \rightarrow \left[\text{STATE} \quad \text{AFF} \quad \left(\text{THING}, \text{THING}, \text{time}, \text{manner} \right) \right]$
- $[\text{STATE_ACTION_TIER}] \rightarrow [\text{STATE}]$
- $[\text{THEMATIC_TIER}] \rightarrow \left[\left(\begin{array}{c} \text{SubordFunc}_1 \left(\left\{ \begin{array}{c} \text{EVENT} \\ \text{STATE} \end{array} \right\} \right), \\ \text{SubordFunc}_2 \left(\left\{ \begin{array}{c} \text{EVENT} \\ \text{STATE} \end{array} \right\} \right), \\ \vdots \end{array} \right) \right]$

⁹Although the arguments of the AFF function in the action tier may be left unspecified (empty) if the agent and/or patient are not specified in the verb semantics.

COMPLEX_DESCRIPTION is divided into EVENT_DESCRIPTION and STATE_DESCRIPTION, each with a different purpose. Event descriptions are used to capture main verb semantics, while the state descriptions may only be used to capture complex properties associated with THINGS. Event descriptions incorporate the standard action tier, while state descriptions can specify any static relationship as the basic relation. Complex nominal properties will be discussed in fuller detail in Section 1.2.9 below.

1.2.4 STATE and EVENT Predicates

Pinker’s four-way functional system has been expanded to seven functions, to incorporate some of the additional functional relations observed by Jackendoff.

The functions can be represented via a three-way featural system based on the two-way system proposed by Pinker (i.e. in figure (1.15)), with minor modifications. The dynamic feature is not interpreted as differentiating STATES and EVENTS, but rather as indicating whether the function involves motion and/or change, or stasis. STATES and EVENTS will then be differentiated with the feature $\pm\text{event}$ ($+\text{event}$ = Event, $-\text{event}$ = State). The control feature is used as proposed by Pinker — to indicate whether the first argument of the function is animate and in control of the event/state or not. The functional system is defined in (1.22).

(1.22)

<i>Predicates</i>	<i>Features</i>		
	Event	Dynamic	Control
GO	+	+	–
STAY	+	–	–
MOVE	+	+	+
ORIENT	–	+	–
BE	–	–	–
HAVE	–	–	+
AFF	–	+	+

The semantic meanings the predicates are intended to convey are listed below. The correspondence of each predicate’s feature triplet to its semantic meaning is specified in *italics*. In order for the intuitive meaning of these predicates to be fully represented, however, it would be necessary to define the inference patterns in which each predicate may appear. This will not be pursued in the current context.

- GO an Event-function which denotes a Thing traversing a Path.
An EVENT which specifies motion of its first argument, which need not be animate.
- STAY an Event-function which denotes stasis over a period of time; two arguments: the Thing standing still and its location (Place).
An EVENT which specifies stasis of its first argument, which need not be animate.
- MOVE an Event-function which specifies that a Thing moves.
An EVENT which specifies motion of its argument, which is generally animate and controlling the motion.
- ORIENT a State-function specifying the orientation of a Thing with respect to a Path.
A STATE which is dynamic in the sense that the orientation of its (generally inanimate) argument is dependent on an external influence which could change. The state is maintained only through the continual action of the external influence.

- BE a State-function for specifying the location (Place) of a Thing.
A STATE which is not dynamic and whose first argument is not in control of the state.
- HAVE a State-function which specifies a Thing which has (possesses) a Thing.
A STATE which is not dynamic, but whose first argument is generally animate and in control of the state of possessing the second argument.
- AFF a State-function which specifies that an actor “affects” a patient.
A STATE which is dynamic in the sense that the first (animate and controlling) argument continously acts upon the second argument for the duration of the state.

In addition, the following functions have been adopted from the Pinker/Jackendoff representations.

- place functions at, on, in, under, ... functions expressing location.
- path functions to, from, toward, away-from, via; functions expressing direction.

A conspicuous gap in the chart in (1.22) is the {+event, −dynamic, +control} triplet. It is difficult to imagine an event function which specifies no change in state yet has an animate, controlling argument. The function would have to be the EVENT function corresponding to the HAVE function, but any likely predicates, such as *receive* or *acquire*, seem better represented with a complex description. The function position will be left unused until there appears to be clear need for a function characterised by those features.

Although Pinker characterises ACT as an event function, the corresponding AFF has been classified as −event because the agent and patient roles are static within the verb semantics. That is, the verb does not specify a *change* in the agentive/patient relations, but rather what those relations are throughout the use of the verb. Thus the action tier reflects a static relation.

The formation rules for the various function predicates in the integrated representation are found below. The argument structures as defined are the only ones allowed in the semantic representations in this work. Note that each of the EVENT and STATE functions is a predicate abbreviation for a {±event, ±dynamic, ±control} triple.

- [EVENT] $\rightarrow \left\{ \left[\begin{array}{l} \text{EVENT GO (THING, PATH, time, manner)} \\ \text{EVENT STAY (THING, PLACE, time)} \\ \text{EVENT MOVE (THING, time, manner)} \end{array} \right] \right\}$
- [STATE] $\rightarrow \left\{ \left[\begin{array}{l} \text{STATE BE (THING, PLACE, time)} \\ \text{STATE HAVE (THING, THING, time)} \\ \text{STATE ORIENT (THING, PATH, time)} \\ \text{STATE AFF (THING, THING, time, manner)} \end{array} \right] \right\}$
- [PLACE] $\rightarrow \left[\text{PLACE place function (THING)} \right]$
- [PATH] $\rightarrow \left[\text{PATH } \left\{ \begin{array}{c} \text{to} \\ \text{from} \\ \text{toward} \\ \text{away – from} \\ \text{via} \end{array} \right\} \left(\left\{ \begin{array}{c} \text{THING} \\ \text{PLACE} \end{array} \right\} \right) \right]$

It was necessary to expand Pinker's four functions into seven mainly due to a desire to maintain a strict argument structure for each function. It can be shown that Pinker's "free" use of argument structure means that while he claims to be using only four basic functions, he is actually using more.

For example, Pinker handles what is expressed by the MOVE predicate by postulating the argument structure in (1.23) for the GO predicate, which differs from his usual argument structure for GO shown in (1.24).

$$(1.23) \quad \left[\text{GO} \left(\text{THING}, \langle \text{manner} \rangle \right) \right]^P$$

$$(1.24) \quad \left[\text{GO} \left(\text{THING}, \text{PATH}, \langle \text{manner} \rangle \right) \right]^P$$

The assignment of the GO predicate to each of the above functions suggests an identity between them, despite their divergent argument structures. Formally, however, when two functions have distinct arities, they cannot be equivalent¹⁰. A common name is meaningful only to the extent that the two functions are related in terms of what they attempt to express. The two uses of GO above are certainly related, in that they both express a particular type of motion, but they differ with respect to their cognitive content. (1.24) expresses a motion of an object towards a destination, for which the focus (or new information) is on the result of the motion, while (1.23) seems to focus particularly on the motion itself. This difference can be observed with respect to the inference patterns of the two functions. Jackendoff proposes (1990, p. 27) the rule in (1.25).

$$(1.25) \quad \begin{array}{l} \text{At the termination of } [\text{GO} (\text{X}, [\text{to} (\text{Y})])], \\ \text{it is the case that } [\text{BE} (\text{X}, [\text{at} (\text{Y})])]. \end{array}$$

Such an inference is not possible in the case of (1.23). Rather, nothing can be inferred about the state of the first argument of GO after completion of the event which the predicate expresses. Furthermore, the two functions seem to differ with respect to the type of their first argument. As indicated by the $\pm\text{control}$ feature represented in (1.22), the MOVE predicate (the equivalent of the GO predicate expressed in (1.23) within the current functional system) takes a controlling or animate argument, while the GO predicate does not require an animate argument. Essentially, MOVE implies that the argument is moving under its own power, while the motion of the GO argument can be due to some external force. It is for these reasons that the two functions are viewed as distinct in this framework.

Pinker's functional system has been augmented with the function STAY in order to provide a mechanism for expressing nonmotional situations. Pinker suggests (1989, p. 183) that this can be handled within his system simply by allowing BE to be a type of EVENT, and that STAY is therefore unnecessary. This treatment requires that his $\pm\text{event}$ feature specify the *unmarked case* of whether the predicate is an EVENT or STATE, rather than specifying a definitive feature of the predicate¹¹. Since the shift from a STATE to an EVENT or vice versa

¹⁰Two functions are equivalent if they have the same mapping from particular inputs to outputs. Clearly this cannot be the case if one function requires two inputs, and the other only one. Although the functions used in this framework are attempting to relate various sorts of objects to one another in specific ways, rather than specifying an output value dependent on input values, they must adhere to the same criteria as all functions.

¹¹Note that again Pinker's system only appears to be restricted to four predicates. The variance introduced by having marked and unmarked cases of a defining feature requires different interpretations of each case of the feature, essentially equivalent to expanding the number of predicates in use. If both of Pinker's features have

involves a change in the interpretation of the predicate, it comes to specify a different relation between the functional arguments in each case. It can therefore be argued that the change in the interpretation of the basic relation (i.e. in the unmarked case) is enough to require a different predicate for each of the basic and changed relations. Even if a new predicate were not used to represent the changed relations, each use would in any case require specification of its meaning (in terms of varying inference patterns). A function would thus have two distinct, though related, interpretations. It seems to be more clear to label each of these interpretations with a different predicate name, while also avoiding having marked and unmarked cases of predicative defining features.

Jackendoff's ORIENT function was adopted in the current framework as it is unclear how Pinker would account for a relation specifying the configuration of an object with respect to another object using his four predicates. None of {GO, ACT, BE, HAVE} seem to capture this type of relationship.

It could be argued that the HAVE predicate is superfluous and that it can be adequately captured by a BE_{possessional} relationship, as in (1.26).

$$(1.26) \quad \text{HAVE}(\text{THING}_1, \text{THING}_2) \iff \text{BE}_{\text{possessional}}(\text{THING}_2, \text{at}(\text{THING}_1))$$

However, Pinker (1989, pp. 189-190) puts forward several arguments which suggest that the HAVE predicate deserves to be a primitive function. They will not be repeated here in full. Pinker's main point is that the verb *have*, represented by a BE_{possessional} predicate, would violate the Thematic Hierarchy Condition¹² in requiring the possessor to be linked to the subject rather than to an oblique object, and the possession to be linked to an object rather than the subject. This representation would mark *have* as an exception to the general relation between thematic roles and syntactic argument structure, despite its frequency and error-free usage by children acquiring verb argument structure¹³. Treating the relation expressed by the verb *have* as primitive maintains the consistency of the Thematic Hierarchy Condition and supports child acquisition data.

Although several of Jackendoff's functions have been eliminated in this representation, it is still possible to achieve the same semantic effects using the existing framework.

- EXT^J *a State-function for the spatial extension of linear objects along a path.*
This function is subsumed by the ORIENT function. A linear object being extended along a path is equivalent to the object being oriented along that path.
- CONF^J *a State-function that expresses that a verb describes the internal spatial configuration of an object; one argument: the Thing that is in the spatial configuration.*
This function is similarly subsumed by the ORIENT function. It can be expressed as [ORIENT(THING₁, to(at(THING₁)))].
- REACT^J *a formal elaboration of an Event which specifies that one Thing reacts to another; two arguments: X, Y; X "reacts to" Y.*

marked and unmarked interpretations for each predicate, then there are actually four possible interpretations for each of the four predicates. Although some of these interpretations would not make any sense with respect to the basic relation which each predicate is intended to capture, it seems more clear to spell out explicitly the possible predicates, the meaning of the defining features, and the precise interpretation of the predicates.

¹²This is the condition which specifies the correspondence of specific thematic roles to syntactic positions.

¹³If *have* were an exception, one would expect that children would reverse the subject and objects of the verb or insert spatial prepositions. This does not appear to be the case.

This function can be expressed using AFF by reversing the order of the arguments. Thus “X reacts to Y” if “Y affects X” in some way. In Jackendoff’s introduction of REACT (1990, p. 137), he points out that REACT is the “mirror image” of AFF. Within his framework, however, it is necessary that REACT exist as a separate function due to his reliance on the argument order in the theory linking semantic argument structure and syntactic argument position.

The remainder of Jackendoff’s functions will be addressed after subordinating functions have been introduced.

1.2.5 A Word on Notation

Before discussing further details of the implemented representation, the notation used in the remainder of the paper will be introduced. A sample lexical entry will be presented, analysed, and related to the equivalent HPSG-style entry.

The figure in (1.27) shows the lexical entry for the verb *pay*. The first line of the entry shows the word to which the entry corresponds, and the second line shows its basic syntactic category. The third line shows the word’s subcategorization list. For verb entries under HPSG, this always includes the subject as the first element. The subsequent arguments on the subcat list appear in surface order (see Section 2.4 for discussion of the subcat list). The remaining lines contain a substructure in which the grammatically relevant semantics is represented. If the verb requires a COMPLEX_DESCRIPTION to capture its semantics, as this example does, the top line of the substructure contains the action tier and the subsequent lines contain the thematic tier. As explained in Section 1.2.2, the thematic tier consists of a set of STATES or EVENTS subordinated by subordinating relations. Each subordinating relation appears in italics (i.e. *effect*), with the subordinated occurrence appearing indented underneath it. Properties are notated as superscripts on THINGS in italics (i.e. *THING₂^{money}*), semantic fields are notated as subscripts in typewriter type on the function to be interpreted in a different field (i.e. *GO_{possession}*), and numerical subscripts notate structure sharing. In particular, the coindexing between NPs and THINGS indicates that they share an INDEX value. The INDEX contains the number, person, and gender information of the noun in the NP, necessary for agreement.

$$(1.27) \quad \left[\begin{array}{l} \text{pay} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2^{\text{money}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_3)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

The implemented HPSG lexical entry from which (1.27) derives appears in (1.28) and (1.29). Nonlocal and contextual information has not been included, as it is not relevant to the problem at hand. Please refer to the type signature, shown in Appendix D.1, for the formal definitions of types of entities and appropriate values for the various features.

$$\begin{array}{l}
 (1.28) \quad \left[\begin{array}{c} \text{PHON} \quad \textit{pay} \\ \text{SYNSEM} \quad \left[\begin{array}{c} \text{LOCAL} \quad \left[\begin{array}{c} \text{CATEGORY} \quad \left[\begin{array}{c} \text{HEAD} \quad \textit{verb} \quad \left[\begin{array}{c} \text{VFORM} \quad \textit{bse} \\ \text{AUX} \quad \textit{minus} \\ \text{INV} \quad \textit{minus} \\ \text{MOD} \quad \textit{none} \\ \text{PRD} \quad \textit{bool} \end{array} \right] \\ \text{MARKING} \quad \textit{unmarked} \\ \text{SUBCAT} \quad \langle \textit{NP}[\text{1}], \textit{NP}[\textit{acc}][\text{2}], \textit{to NP}[\textit{acc}][\text{3}]\rangle \end{array} \right] \\ \text{CONT} \quad \left[\begin{array}{c} \text{NUCLEUS} \quad [\text{8}] : (\text{see (1.29) below}) \\ \text{QUANTS} \quad \textit{e_list} \end{array} \right] \end{array} \right] \end{array} \right] \end{array} \right]
 \end{array}
 \end{array}$$

$$\begin{array}{l}
 (1.29) \quad [\text{8}] : \left[\begin{array}{c} \text{PROP} \quad \langle [\text{4}] : [\text{INDEX} \text{ [1] } \text{RESTR} \textit{e_set}], [\text{5}] : \text{RESTR} \left\{ \begin{array}{c} \text{INDEX} \text{ [2]} \\ \left[\begin{array}{c} \text{NUCLEUS} \quad \left[\begin{array}{c} \textit{money} \\ \text{INST} \quad [\text{2}] \end{array} \right] \\ \text{QUANTS} \quad \textit{list_quant} \end{array} \right\} , [\text{6}] : [\text{INDEX} \text{ [3] } \text{RESTR} \textit{e_set}] \rangle \\ \text{ACTION} \quad \left[\begin{array}{c} \text{AFUNC} \quad \textit{aff_func} \\ \text{ARG1} \quad [\text{4}] \\ \text{ARG2} \quad [\text{5}] \\ \text{MANNER} \quad \textit{no_manner} \\ \text{SEM_FIELD} \quad \textit{sem_field} \\ \text{TIME} \quad [\text{7}] : \textit{time_0} \end{array} \right] \\ \text{STRUCT} \quad \left[\begin{array}{c} \text{THEMATIC} \quad \left\{ \begin{array}{c} \text{AFUNC} \quad \textit{effect} \\ \left[\begin{array}{c} \text{AFUNC} \quad \textit{go_func} \\ \text{ARG1} \quad [\text{5}] \end{array} \right] \\ \left[\begin{array}{c} \text{AFUNC} \quad \textit{to_path} \\ \text{ARG1} \quad \left[\begin{array}{c} \text{AFUNC} \quad \textit{at_place} \\ \text{ARG1} \quad [\text{6}] \end{array} \right] \end{array} \right] \\ \text{MANNER} \quad \textit{no_manner} \\ \text{SEM_FIELD} \quad \textit{possession} \\ \text{TIME} \quad [\text{7}] \end{array} \right\} \end{array} \right] \\ \textit{complex_sem} \end{array} \right]
 \end{array}$$

The phonological (PHON) feature has as its value the written word to which the entry corresponds, for lack of a more precise phonological transcription. The SYNSEM feature contains the syntactic and semantic information associated with the word being represented. Only the LOCAL information is relevant, specifically the CATEGORY and CONT (content) information. CATEGORY includes the HEAD features of the verb, all of the features defining the form of the verb and how it can be used (AUX specifies whether the verb is an auxilliary verb, INV specifies whether the verb can appear in inverted form, MOD contains verb modifier information, PRD specifies whether the verb is predicative). The CATEGORY also contains a MARKING feature which indicates whether the verb is being used within a complementized clause (see Pollard and Sag 1994, pp. 45-47), and the SUBCAT feature which has a list of

synsem objects, corresponding to the SYNSEM values of the signs with which the verb must combine to become “saturated”.

The CONT (content) field has two features: the NUCLEUS, containing the core of the semantic information, and QUANTS, used in the HPSG treatment of quantification (see Pollard and Sag 1994, ch. 8). The value of the nucleus field in this work differs dramatically from what appears in Pollard and Sag’s original HPSG work. It is where the semantic representation described in the previous sections is integrated into HPSG.

The PROP (properties) feature is a list of *nom_obj* entities, used in the enforcement of selectional restrictions (discussed in Section 1.2.9). The INDEX feature of each *nom_obj* in the PROP list is structure-shared with the index of one of the NPs in the SUBCAT list, and the RESTR features is used to specify any of the selectional restrictions the verb imposes on that NP. Thus in the example in (1.28) and (1.29), *pay* requires its direct object (the argument after the subject on the SUBCAT list) to have the property *money*.

The value of the STRUCT feature is the semantic structure of the verb. The example shown above shows a complex semantic structure, consisting of both an action and a thematic tier. The complex structure is far more common than a simple structure, and in fact the only kind of structure used in the lexicon of the current implementation. The value of the ACTION feature is an *aff_state* entity, which specifies the AFF function as the main function, and restricts the two arguments of this function to be of *things*. Additionally, the features MANNER, SEM_FIELD, and TIME are specified for this function. The THEMATIC feature has as its value a set of *thematic* entities, each of which must have a subordinating relation as the value for AFUNC, and an occurrence as the value of the subordinating function’s ARG1. The occurrence can specify any type of function and its arguments. In the example, there is only one subordinated event in the thematic tier, and it is a *go_event* entity, specified for MANNER:*no_manner* and SEM_FIELD:*possession*, subordinated by the *effect* subordinating function. Other verbs with more subordinated occurrences simply will have more elements specified in the THEMATIC set.

1.2.6 Subordinating Functions

Subordinating functions are used to express complex verb semantics by relating subevents within verb semantic structure in particular ways. These functions are “subordinating” because their arguments are EVENTS or STATES which are not the main event expressed by the verb, but rather are related to the main event in some way. They constrain the semantic structure of a verb to convey a specific meaning.

The sextuple of features introduced by Pinker (see Section 1.1.2) for characterising subordinating relations will be adopted here, as well as the mnemonics he uses to clarify what the relation is intended to convey. Also following Pinker, it will be possible to specify a set of events which are subordinated to the main event in different ways. This is implemented as a set of subordinated functions and their arguments in the thematic tier.

Some verbs will have simple structure, while others will require many subordinated relations to represent their full semantics. An example of a verb which requires only simple semantic structure is *roll*, in the sense of *The ball rolled*. Its representation can be found in example (1.30).

$$(1.30) \quad \left[\text{MOVE} \quad (\text{THING}^{ball}, \text{rolling_manner}) \right]$$

One of the more complicated structures is the representation of *buy*, in the sense of *John buys a book for Mary*, shown in (1.31). This representation can be paraphrased as “John affects a book such that the book goes into John’s possession, in order for Mary to have the book. John’s action with respect to the book obligates him to affect money such that it goes into someone/something else’s possession.”

$$(1.31) \quad \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to (at (THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF (THING}_1, \text{THING}_4^{\text{money}}, \text{time}_2, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_4, \text{to (at (THING)), \text{time}_2, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Using subordinating functions, Jackendoff’s functions not addressed in Section 1.2.4 can also be handled.

- CAUSE^J *an Event-function specifying cause and effect relations; two arguments: a Thing which is the agent or an Event which is the cause, and an Event which is the effect.*

This can be represented using the subordinating function *effect*, and specifying the agent in the action tier. Thus *butter*, as in *Harry₁ buttered the bread₂*, represented by Jackendoff (1990, p. 54) as (1.32a), would be represented as (1.32b).

$$(1.32) \quad \begin{array}{ll} \text{a.} & [\text{EVENT CAUSE (THING}_1, [\text{GO (THING}^{\text{butter}}, \text{to (on (THING}_2)))])]^J \\ \text{b.} & \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_3^{\text{butter}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO (THING}_3, \text{to (on (THING}_2)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array}$$

- INCH^J *an Event-function specifying the “inchoative”, a change taking place which has a final state; one argument: the State in which the event terminates.*

This can also be represented using the subordinating function *effect*, but requires an unspecified action tier (that is, the arguments of AFF will not be linked to any syntactic arguments). Thus *point*, as in *The weathervane₁ pointed north₂*, represented by Jackendoff (1990, p. 92) as (1.33a), would be represented as (1.33b).

$$(1.33) \quad \begin{array}{ll} \text{a.} & [\text{EVENT INCH ([ORIENT (THING}_1, \text{PATH}_2))}]^J \\ \text{b.} & \left[\begin{array}{l} \text{AFF (THING, THING, time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{ORIENT (THING}_1, \text{PATH}_2, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array}$$

- EXCH^J *a modifying function specifying an event which is “in exchange for” the entity being modified.*

This function corresponds to the subordinating function *obligates*. Thus *buy*, as in *John₁ buys a book₂ for Mary₃*, which would be represented by Jackendoff as (1.34), would be represented in the current system as (1.31)¹⁴.

¹⁴Note that in Jackendoff’s representational system, it is unclear how to indicate that the book is intended to be given to Mary.

$$(1.34) \quad \left[\begin{array}{c} \text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1))) \\ \text{EVENT} \quad [\text{EXCH} [\text{GO}_{\text{possession}} (\text{THING}^{\text{money}}, \text{from} (\text{THING}_1))]] \end{array} \right]^J$$

Like Jackendoff’s main predicates, his subordinating relations are also subsumed by other mechanisms within the current framework.

- BY^J *X is the means to the accomplishment of Y.*
Corresponds to the subordinating function *means*.
- FROM^J *X is the reason why Y occurs.*
Corresponds to the subordinating function *cause*.
- FOR^J *X is the intended result of the action in Y.*
Corresponds to the subordinating function *for_to*.
- WITH^J is used in several ways in the Jackendoff approach.
 - In the instrumental sense, WITH corresponds to the subordinating function *means*.
 - Jackendoff represents *Bill ate the meat raw* as in (1.35) (1990, p. 202). This use of WITH can simply be represented by adding property *raw* to *meat*¹⁵.

$$(1.35) \quad \left[\begin{array}{c} \text{EAT} (\text{bill}, \text{meat}) \\ \text{EVENT} \quad [\text{WITH} [\text{BE} (\text{meat}, \text{at} (\text{raw}))]] \end{array} \right]^J$$

- Jackendoff represents *Bill entered the room smiling* as in (1.36) (1990, p. 98). This use of WITH can be represented by specifying the manner of the GO predicate as *smiling*.

$$(1.36) \quad \left[\begin{array}{c} \text{GO} (\text{bill}, \text{to} (\text{in} (\text{room}))) \\ \text{EVENT} \quad [\text{WITH} [\text{SMILE} (\text{bill})]] \end{array} \right]^J$$

1.2.7 Manner

Pinker (1989, p. 174) introduces **manner** as a recurrent element of verb meaning indicating how an actor acts or a theme changes. For example, *X kisses Y* is represented as “X acts upon/affects Y in a *kissing* manner”. Only certain of the predicates introduced above can incorporate such a manner specification. Clearly *Affecting* can be done in a particular manner. Similarly, *GOing* and *MOVE-ing* can be done in a certain manner. On the other hand, it does not make sense to *STAY*, *HAVE*, *ORIENT*, or *BE* in a particular way. Thus **manner** is an argument only of the functions for which it makes sense. For those functions, however, it is an element which must be specified — either as *no-manner*, or as a particular manner. It is not an optional argument.

1.2.8 Time

The temporal notation adopted here has not been designed in an attempt to capture the full range of verb temporal distribution and aspect. Rather, it is the minimal representation required to model distinctions specifically relevant to the dative alternation. Temporal relations among subevents in the verbal semantic structure are represented through the association of specific “time points” — time_0 , time_1 , time_2 — with each occurrence. Occurrences with no clear endpoint are indicated by a *continuous* marker in the place of a time point. Thus all

¹⁵Clearly in the current system EAT is not a permissible function. Assume that it is a mnemonic for a more complicated structure, along the lines of $[\text{GO} (\text{meat}, \text{to} (\text{in} (\text{Bill})), \text{time}, \text{eating_manner})]$.

STATE and EVENT functions take an additional *time* argument. An example of the use of the time points is found in the lexical entry for the verb *send* in (1.37)¹⁶. Here, the three related occurrences occur at distinct points in time.

$$(1.37) \quad \left[\begin{array}{l} \text{send} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner)} \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to (at (THING}_3)), \text{time}_2, \text{no_manner})), \\ \text{means} \\ (\text{AFF (THING}^{\text{postal service}}, \text{THING}_2, \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

1.2.9 Properties

Both Jackendoff and Pinker introduce PROPERTY as a basic semantic conceptual category. Properties are characteristics which can be attributed to particular things. Jackendoff relies mainly upon a 3D model for representation of the physical properties of objects, but he treats properties which appear in a sentence (such as adjectives) as modifiers of type PROPERTY, as shown in example (1.11). Pinker, on the other hand, focuses on representing grammatically relevant properties. He defines a hierarchy of properties, introduced in Section 1.1.2 and repeated in (1.38), which are systematically used in verb argument specification and which can affect syntax¹⁷. In addition, he allows for verb selection of highly specific kinds of arguments through reference to generic properties that the argument must have; these particular properties, however, cannot provide a basis for grammatical differences.

$$(1.38) \quad \text{PROPERTY} \rightarrow \left(\begin{array}{l} \text{animate} \left(\begin{array}{l} \text{human} \\ \text{nonhuman} \end{array} \right) \\ \text{inanimate} \left(\begin{array}{l} 0D \\ 1D \\ 2D \\ 3D \end{array} \right) \\ \text{count} \left(\begin{array}{l} \text{rigid} \\ \text{flexible} \end{array} \right) \\ \text{mass} \left(\begin{array}{l} \text{substance} \left(\begin{array}{l} \text{liquid} \\ \text{semisolid} \end{array} \right) \\ \text{aggregate} \rightarrow \begin{array}{c} \text{Parts} \\ | \\ \text{PROPERTY} \end{array} \end{array} \right) \end{array} \right)$$

Complex properties exist in addition to simple ones. These can be defined in terms of states. An example comes from Pinker, p. 187 – “X covered with snow” can be represented as “X such that [_{STATE} BE (snow, (on (X)))]”. Here, the relation which subordinates the state is *such_that*. Another type of complex property is one which specifies the intended use or function of the object with that property. The subordinating relation for this type is *for_to*.

¹⁶The use of the notation $\text{THING}^{\text{postal service}}$ will be explained in Section 1.2.9.

¹⁷For discussion, see Pinker 1989, specifically in his explanation of the locative alternation, pp. 228-239.

The arguments of these complex property relations are STATE_DESCRIPTIONS, as described in Section 1.2.3.

Two ways of handling properties will be considered. One method is to associate a list of properties with something, and another is to develop a complex taxonomy of types, in which things are subtypes of more general types and inherit the properties of these supertypes. For example, *water* can either be represented as [WATER, properties: {liquid, ...}], or as a subtype of the type LIQUID. Selectional restrictions are enforced in the first instance by searching for a particular property in the list, and in the second instance by requiring an object to be of a particular type.

Jackendoff seems to prefer a view of objects within a taxonomy of conceptual constituents. To that end, he proposes a rule of *Argument Fusion*, in which the type of an argument must be fused with the type of the selectional restriction on the argument as indicated by the verb semantics. Fusion is impossible if there is any incompatible information in the types, such as if they appear as mutually exclusive possibilities in the type taxonomy.

Pinker chooses a treatment of selectional restrictions different from that of Jackendoff. He says, “a semantic representation for a verb can impose selection restrictions on its arguments by listing exactly one label from any of the parenthesised sets in [(1.38)]”. Since there are multiple compatible parenthesised sets, each selectional restriction may actually be a list of properties.

The current implementation associates a list of properties with each referential element, using an attribute of the SYNSEM:CONTENT component of the HPSG representation. This attribute is called the RESTRICTION, and is only associated with nominal objects. A THING is defined in the type signature (as modified for the current implementation) to be a nominal object, and thus has this attribute. The value of the RESTRICTION attribute is a set of restrictions, or semantic conditions, anchored to specific entities (referenced by their INDEX values). An example of the use of this restriction to specify the semantics (SYNSEM) of the word *book* is shown in (1.39)¹⁸. Properties, then, are treated as semantic conditions on the entities to which they refer. These properties may come from the set in (1.38), but many more are highly specific characteristics not relevant to syntax but important for describing selectional restrictions. Since properties are semantic conditions, they are represented just like any such restrictions. This is shown in (1.40), in which a THING is given a *consumable* property.

$$(1.39) \quad \left[\begin{array}{c} \text{SYNSEM: LOCAL: CONTENT} \\ \left[\begin{array}{c} \text{INDEX } \boxed{1} \left[\begin{array}{c} \text{PER } 3rd \\ \text{NUM } sing \\ \text{GEND } neut \end{array} \right] \\ \text{RESTR } \left\{ \left[\begin{array}{c} \text{RELATION } book \\ \text{INSTANCE } \boxed{1} \end{array} \right] \right\} \end{array} \right] \end{array} \right]$$

¹⁸Note that in this attribute-value matrix the complex structure actually used in the implementation to handle quantification properly has been simplified for clarity.

$$(1.40) \quad \left[\begin{array}{c} \text{SYNSEM: LOCAL: CONTENT} \\ \left[\begin{array}{c} \text{INDEX } \boxed{1} \left[\begin{array}{c} \text{PER } per \\ \text{NUM } num \\ \text{GEND } gend \end{array} \right] \\ \text{RESTR } \left\{ \begin{array}{c} \text{RELATION } consumable \\ \text{INSTANCE } \boxed{1} \end{array} \right\} \end{array} \right] \end{array} \right]$$

The notation for representation of properties attached to a **THING** constituent in this document is simply to superscript the property to the constituent. Thus the representation for the constituent in (1.40) would be **THING**^{consumable}.

Just as verbs can impose selectional restrictions on their arguments, prepositions can also imposed conditions on their objects. For example, the preposition *with*, in the accompaniment sense (as in “I went to the store *with Mary*”) requires an animate object. This has implications for semantic interpretation relevant to the current context, in particular for the verbs *reward*¹⁹ and *honor*. Both of these verbs can appear in the same basic syntactic patterns: “X rewards/honors Y with Z” and “X rewards Y”. These two forms are captured in distinct lexical entries, as in (1.41).

$$(1.41) \quad \begin{array}{ll} \text{a.} & \left[\begin{array}{c} \text{reward} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{with NP}_2 \\ \left[\begin{array}{c} \text{AFF (THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner)} \\ \left\{ \begin{array}{c} \text{effect} \\ \left(\text{HAVE (THING}_3, \right. \right. \\ \left. \left. \left\{ \begin{array}{c} \text{for_to (BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ \text{EVENT (THING}_3) \end{array} \right\}, \right\} \right) \end{array} \right\} \\ \text{THING}_2 \\ \text{time}_0) \end{array} \right] \end{array} \right] \\ \text{b.} & \left[\begin{array}{c} \text{reward} \\ \text{V} \\ \text{NP}_1, \text{NP}_3 \\ \left[\begin{array}{c} \text{AFF (THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner)} \\ \left\{ \begin{array}{c} \text{effect} \\ \left(\text{HAVE (THING}_3, \text{THING}_2^{\text{reward}}, \text{time}_0) \right) \end{array} \right\} \end{array} \right] \end{array} \right] \end{array}$$

The selectional restrictions imposed by prepositions become important when the system is attempting to interpret sentences such as those in (1.42).

- (1.42) a. Mary rewards her son with candy.
b. Mary rewards her son with a bike.

¹⁹This verb is a good example of verbs imposing selectional restrictions which are difficult to characterise. For the lexical entry (1.41a) the object of the preposition must be something that could be considered a reward. This discussion will be taken up later in the current section.

An interpretation based on (1.41a) will represent the semantics of this sentence appropriately. However, it is also possible for the system to use the lexical entry in (1.41b), treating the prepositional phrase as an adjunct. The preposition *with* can be interpreted as an adjunct in several ways – in the accompaniment sense (*I went to the store with Mary*), in the instrument sense (*I cracked the egg with a knife*), and in the means sense (*I went to school with the bus*). Each of these senses require an object that is compatible with a particular cognitive notion – animate, instrument, and means, respectively. The object of *with* in (1.42a), *candy*, does not seem to be compatible with any of these notions. Thus if the selectional restrictions are properly enforced, any semantic interpretation for this sentence treating the prepositional phrase as an adjunct will be ruled out. On the other hand, a “bike” is certainly compatible with the notion of being a means (as in *I went to school with my bike*). Therefore, although (1.42b) is clearly not ambiguous for humans, it will be ambiguous to the system because there is no obvious way to enforce the idea that *reward* cannot have a means adjunct associated with it²⁰.

The syntactic alternation under consideration in this work, the dative alternation, does not depend on selectional restrictions. As a result, any selectional restrictions which appear in the semantic structure of the verbs represented in the implementation do not fall within the system of grammatically relevant properties found in (1.38). Furthermore, most of the properties are attached to things which are not syntactic arguments. Rather, they are attached to things which play some role in the semantics of a verb, but which do not appear in the syntax of the sentence. In this case, the properties are not used as selectional restrictions, but instead provide information about characteristics of something important in the semantic description of the verb. For example, when *X sends Y to Z*, “X affects Y such that Y goes to Z, by means of the *postal service* acting upon Y”. This is represented as in (1.43).

$$(1.43) \quad \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to (at (THING}_3)), \text{time}_2, \text{no_manner})), \\ \text{means} \\ (\text{AFF (THING}^{\text{postal service}}, \text{THING}_2, \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right]$$

Here *postal service* serves a function in explicitly defining the semantics of *send* but does not appear as a syntactic argument of the verb. The representation of *postal service* is analogous to that of *book* in (1.39) – it is an object with specific constraints on its interpretation.

The mechanisms of HPSG create some difficulty for the enforcement of selectional restrictions. Jackendoff’s rule of Argument Fusion is realised in HPSG through the use of the Head Feature Principle combined with unification of coindexed positions (structure sharing). It is not possible to indicate any operation at the level of a lexical entry other than structure sharing of its component parts. However, structure sharing is not applicable to the handling of selectional restrictions. There are two property lists involved in handling these restrictions — the property list imposed by the verb on its argument and the property list associated with the realisation of the argument. The operation of matching constraints imposed by the verb on its arguments with the properties of the actual arguments cannot be accomplished through simple unification of the two property lists, because the realised argument will have many more properties than will be required by the selectional restrictions of the verb. Accommo-

²⁰This could in theory be done by subdividing verbs into classes that can take particular kinds of adjuncts, but this would seem to create artificial distinctions between verbs.

dating this more complicated operation would involve either adding a principle or modifying an existing principle to handle “matching” at the time of syntactic and semantic integration.

The question remains as to precisely how the matching operation would be implemented. One possibility is to implement it as a subset operation. In particular, the set of constraints imposed by the verb on a noun phrase would be required to be a subset of the actual properties of the noun phrase. In addition to being expensive computationally²¹, this method is inadequate for modelling the cognitive approach to matching. Consider, for example, the lexical entry of the verb *prefer* in (1.44).

$$(1.44) \left[\begin{array}{l} \text{prefer} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2^{\text{preferred}}, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{cause} \\ (\text{AFF}^+ (\text{THING}_2, \text{THING}_1, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

The semantics of *prefer* require its argument to be a *preferred* object. This is obviously an attribute of the object which depends on the context in which the sentence containing *prefer* is uttered, and the mental representation of the situation. What is really needed is not a subset operation but rather a “compatibility” operation which incorporates contextual factors into the determination of whether a particular instantiation of an argument satisfies the selectional restrictions imposed by the verb.

Another compatibility issue which arises with respect to the use of properties in the representation is that of complex properties. The *for-to* complex properties indicate some intended use of the object, rather than a true characteristic of the object. For example, in the semantic structure of the verb *present* in (1.45), the complex property indicates that THING_2 which is being presented to THING_3 is supposed to be THING_3 ’s because of some event that involved THING_3 . This property therefore indicates why the particular THING_2 is being presented to THING_3 .

$$(1.45) \left[\begin{array}{l} \text{present} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{with NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ \left(\begin{array}{l} \text{GO}_{\text{possession}} (\text{THING}_2 \\ \text{to} (\text{at} (\text{THING}_3)), \text{time}_0, \text{no_manner}) \end{array} \right) \text{for-to} \left(\begin{array}{l} \text{BE}_{\text{possession}} (\text{THING}_2, \text{at} (\text{THING}_3)) \\ \text{fulfills} \\ (\text{EVENT} (\text{THING}_3)) \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Determination of whether a particular object satisfies the complex property again requires knowledge of the context in which the utterance occurs, and other bits of information such as the awards that a particular event has associated with it. For example, in *Bob presented Sue with the medal*, the medal must be the prize associated with some event in which Sue acted in such a way as to deserve the medal. Establishing whether a particular sentence has

²¹The subset operation requires taking each element from the first set in sequence, and then searching for a matching element in the second set, which is also done sequentially. This is therefore $O(n^2)$ in the worst case, where n is the number of elements in a set.

a meaningful interpretation relevant to contextual conditions is clearly an extremely complicated task. The solution to this problem is quite important, as selectional restrictions are fundamental in semantic representation. The implementation used in this work, for example, contains function predicates built from several features, one of which is \pm control. A +control function must have a first argument which is in control of the event/state. This would seem to require a selectional restriction to be enforced through the type hierarchy. It may suffice to require the first argument to be +animate, but if there is no way to verify such a requirement, the \pm control feature is essentially meaningless.

Many issues have been raised in the above discussion. The current implementation only addresses the most fundamental of the issues. A principle has been added to the HPSG principles (introduced in Pollard and Sag 1994) specifically to handle selectional restrictions imposed by a verb on its arguments. The matching operation is treated as a subset operation for simplicity. The relevant ALE code is found in (1.46). This code depends on the PROP list in the semantic representation of a verb as introduced in Section 1.2.5. Each NP or PP element on a verb's SUBCAT list must have a corresponding element on the PROP list. These PROP elements must appear in the same order as they appear on the SUBCAT list. The elements on the PROP list specify any properties which the verb requires the corresponding NP²² to have (the selectional restrictions imposed by the verb on that argument).

The principle in (1.46) is one of the principles which the HPSG schemata must satisfy. Thus, the principle is called as a goal in the grammar rules defining the schemata for which it is appropriate. It is called with the sign for the head daughter in the schema and the list of the SYNSEM values of the complement daughters (see Pollard and Sag 1994, ch. 1 for a discussion of the schemata). The complement daughters will either correspond to the entire SUBCAT list of the verb, or the SUBCAT list minus the subject (the first element). The predicate `sre_principle_match` is responsible for determining which of these alternatives is in fact the case in any given application of the principle, and for picking out the corresponding parts of the PROP list. `sre_principle_help` then recurses down the list of SYNSEMs and the PROP list, picking off the respective RESTR values for corresponding elements, and checking to make sure that the set of required RESTR elements as specified in the PROP list is a subset of the set of RESTR elements actually instantiated on the verb argument. This is done until there are no more elements on the SUBCAT list, ensuring that all selection restrictions are satisfied.

Solutions for imposition of selectional restrictions by prepositions or the “cognitive compatibility” problems have not been provided in this work. However, the solution of the basic problem of verb imposition of selectional restrictions provided shows that the selectional restrictions can be handled to a certain extent within the HPSG framework. Further work needs to be done on these issues.

²²Restrictions on PP arguments are interpreted as restrictions on the NP object of the preposition.

```

%% sre_principle(+HeadDtr, +Comp-Dtr-Synsems)
%%
sre_principle((synsem:(loc:((cat):(head:verb,
                                subcat:Subcat),
                                cont:nucleus:(top_level_desc,
                                                prop:Proplist)))),
              Comps) if
  !, sre_principle_match(Subcat, Proplist, Comps).

sre_principle(_,_) if true.

%% sre_principle_match(+Subcat, +Property-List, +Comp-Dtr-Synsems)
%%
sre_principle_match(Subcat, Proplist, Subcat) if
  sre_principle_help(Subcat, Proplist).

sre_principle_match([_H|Comps], [_HP|TP], Comps) if
(1.46)   sre_principle_help(Comps, TP).

%% sre_principle_help(+List-Synsems, +Corresponding-Property-List)
%%
sre_principle_help([],_) if
  true.

sre_principle_help([(loc:((cat):(head:(prep;noun),
                                subcat:[]),
                                cont:(index:Ind,
                                        restr:Restr)))|RestSubcat],
                  [(index:Ind,restr:ReqRestr)|RestProp]) if
  subset(ReqRestr, Restr),
  sre_principle_help(RestSubcat, RestProp).

sre_principle_help([(loc:((cat):(head:(func;adj;verb;reltvzr))))
                  |RestSubcat],
                  Proplist) if
  sre_principle_help(RestSubcat, Proplist).

```

1.2.10 Semantic Field specification

The predicates introduced in (1.22) have relevance to spatial domains. It is, however, clear that the predicates and the associated machinery can be extended to nonlocational semantic fields. These extensions must be specified in two ways. First, an interpretation scheme must be provided to indicate how a spatial predicate maps to a particular relational notion within the new semantic field. Second, constraints on the types or properties of constituents which can serve as the arguments to the predicates must be indicated.

The use of spatial predicates in new semantic fields doesn't require completely new inference patterns. The inference patterns for the spatial predicates may only have to be slightly adjusted to fit the new semantic field. This differentiates the interpretation shift between semantic fields from the shift required to interpret a STATE predicate as an EVENT and vice versa, discussed in Section 1.2.4, for which the inference patterns vary significantly. As a result, it is justified to treat predicates with a semantic field annotation as a variant of the basic spatial predicates rather than postulating an entirely new set of predicates for each field. This is particularly true since the main element which varies between semantic fields is the relation that assumes the role played by location. Thus the adjustments in interpretation of the semantic field depend mainly on only one factor which is easily specified.

Pinker and Jackendoff postulate different semantic fields. Jackendoff defines his fields much more precisely than does Pinker, but since Pinker's analysis of the lexical rules needed to handle syntactic alternations is the basis for the lexical rules in this implementation, Pinker's fields will be adopted.

The semantic fields which Jackendoff proposes (Jackendoff 1983, Chapter 10) are found below. He justifies them by illustrating how they apply to various verbs in a range of functional possibilities.

- **temporal** Times appear as reference objects and the role of location is played by the time of occurrence.
- **possession**
 - **alienable** Possession which is not necessarily permanent. The role of location is played by the relation of being alienably possessed. Thus “Y possesses X” is parallel to “X is at Y”.
 - * **ownership** The role of location is played by the relation of owning (“X is at Y” = “Y owns X”).
 - * **temporary control** The role of location is played by the relation of temporary control.
 - **inalienable** Possession which is necessarily permanent, as in the way one possesses one's head. The role of location is played by the relation of inalienable possession.
- **identificational** Concerns categorisation and the ascription of properties. The role of location is the relation of being an instance of a category or having a property.
- **circumstantial** Events or states appear as reference objects and “X is a character of the main clause Y” plays the role of “X is at Y”.
- **existential** The only possible reference region for location is “existence”.

The semantic fields which Pinker introduces (Pinker 1989, Chapter 5) in addition to Jackendoff's temporal, identificational, existential, and epistemic fields are as follows:

- **possessional**
 - **physical custody** The role of location is played by the relation of being in physical control of an object.
 - **communication** The role of location is played by the relation of mentally storing ideas; Going indicates a transferral of an idea to a new place (or mental storage area).

- **physical** This field specifies that the nature of the relationship which is to be interpreted in this field must be a physical one, i.e. involving particular physical properties such as colour, sound, etc.
- **perceptual** This field is the counterpart to the **physical** field – if something is affected in a physical way, then a thing can *perceive* the physical properties (particularly, the **THING** that is affected in a physical way can **GO** to a place in a perceptual way; the place will perceive the **THING**).
- **epistemic** Ideas appear as reference objects and the role of location is played by the possessor, or the mind containing the ideas (mainly used for **BE** states).

The above semantic fields apply mainly to **GO**- and **BE**- type **EVENTS**. Pinker also uses semantic fields as a way of providing a new interpretation for various **STATE** relationships.

- **social** The acting relationship is construed as social pressure in these verbs.
- **intrapsychic** Different parts of the mind are viewed to be “pitted against one another”.
- **responsibility** Expresses relations among states that are asymmetrically responsible for the existence of other states.
- **psychological**
 - When the agent is a “represented” entity and the patient “animate”, the **ACT** relationship can be interpreted as indicating that a perceived object, event, or idea impinges upon the perceiver.
 - When the agent is “animate” and the patient “represented”, some mental activity or state of the perceiver is responsible for the idea standing in some relation to the mind of the perceiver.

1.2.11 Summary of Differences

- **Grammatically Irrelevant Semantic Content**

Due to Jackendoff’s desire to identify a set of primitives adequate to model all conceptual knowledge, he does not distinguish between *grammatically relevant* semantic content and *grammatically irrelevant* semantic content. Pinker, in contrast, clearly distinguishes between the two, focusing not on representation of conceptual knowledge but only on representation of the subset of conceptual knowledge which has relevance to syntax.

In the representations resulting from these differing approaches, there are few obvious differences. This is because Jackendoff relies on linguistic data for his analysis, from which he mainly has access to syntactically relevant semantics. The differences which do exist appear in the treatment of unelaborated conceptual information. Pinker is not concerned with how a particular **THING** or **PROPERTY** is represented, he is simply concerned that an argument *is* a **THING** or *has* an associated **PROPERTY**, because this information has consistent syntactic implications. He therefore relies on “pointers” to representations of conceptual information for incorporation of grammatically irrelevant semantic content, in the form of quoted constants. Jackendoff in fact does not provide the conceptual representation for such information either, but he uses constants to indicate that a decomposition for the constituent must exist. In some cases, Jackendoff does not even provide a reference to the grammatically irrelevant semantic content, and he does not discuss how the additional elements of meaning would be incorporated.

In the current implementation, Pinker’s approach is adopted. The content of properties or manners is not important, but is simply referenced via an English word. In particular, this is accomplished through the use of the `RESTRICTION` set on nominal objects, as discussed in Sections 1.2.5 and 1.2.9.

- **Action and Thematic Tiers**

Jackendoff utilizes two tiers of conceptual information, supplemented with additional levels of subordinated relations, to show a clear division between types of role info in verb semantics. Pinker does not deem this division to be an accurate reflection of inter-verb relations and formally eliminates it. However, as discussed in Section 1.2.2, the relations expressed by these tiers are still present in Pinker’s analysis.

The current implementation incorporates both an action and a thematic tier, but restructures Jackendoff’s thematic tier to be a set of subordinated relations. This restructured tier allows for a clearer indication of the relationship between multiple occurrences specified in a verb’s semantics.

- **Functions**

Jackendoff and Pinker differ as to which functions are considered primitive. This implementation is able to accommodate the functions proposed by each of them utilizing a featural system based on that used by Pinker.

Jackendoff also allows the elaboration of existing predicates via features to specialize the meaning associated with the predicates, while Pinker does not introduce this mechanism. The only such featural elaboration used in this implementation is a polarity feature on the function `AFF`. This is used in the representation of the verbs of choosing, to suggest that something positively affects something else. In the representation of other verbs, however, this feature is not specified and should therefore be interpreted as “affects in some way”. Such features do not seem to be directly relevant to syntax, though, and are probably not necessary in the current implementation. They are purely a mechanism for capturing a bit more precisely the meaning of a verb.

- **Subordinating Relations**

Pinker introduces a featural system for decomposition of causal links. He justifies it by citing evidence that children’s errors correspond to underdeveloped feature specifications of the links.

Since all of Jackendoff’s subordinating relations can be captured within Pinker’s system or via some other aspect of the representation, Pinker’s system was adopted in the current implementation.

- **Semantic Fields**

Jackendoff and Pinker both subscribe to the Thematic Relations Hypothesis, and extend interpretation of the basic predicates to new semantic fields. Jackendoff associates the semantic field with the predicate, while Pinker associates it with the `STATE` or `EVENT` being described in the new field. The current implementation adopts Jackendoff’s approach, associating the semantic field with the predicate, since it is the interpretation of the predicate which is affected by the shift to a new field.

The differences in Jackendoff and Pinker’s choices for semantic fields will not be discussed in this context. Pinker’s semantic fields relevant to the dative alternation have

simply been adopted because they are critical in his analysis of this alternation. Any semantic field which is not explicitly specified should be assumed to indicate a **spatial** field.

- **Lexical Rules vs. Adjunct Rules**

Although Jackendoff discusses the differing semantic effects of various syntactic forms, he does not propose any formal method for capturing the relationship between varying syntactic constructions involving one verb. He does, however, show that certain elements in particular syntactic constructions can have systematic semantic contributions and attempts to capture these via adjunct rules.

Pinker, on the other hand, does not distinguish between arguments and adjuncts, relying instead upon a representation which idiosyncratically incorporates the contribution of each element in a sentence. It is possible to make generalisations over these contributions, but this is not a formal component of his theory. It is only incorporated insofar as all verbs appearing in a particular syntactic environment share a common semantic structure, and by lexical rules which require a verb to have a particular semantic structure in order to participate in a syntactic alternation. Not only do lexical rules capture the semantic requirements for syntactic alternation, they also formalise its semantic effects.

Lexical rules are the mechanism favoured in the current framework for modelling semantic effects on syntax. Jackendoff's adjunct rules have not been defined precisely enough to incorporate all of the semantic criteria relevant to syntactic alternations.

Prepositional phrase adjuncts can be parsed in the current implementation via the mechanisms of HPSG, but as of yet there is no way to integrate their semantic contributions with the semantics of the remainder of the sentence (as governed by the verb semantics).

The differences in the three approaches become clear when looking at lexical entries and the application of the representations to sentences.

First consider the lexical entry for the dative form of the verb *give*. This highlights the structural differences between the three representations. Pinker's representation is in (1.47), Jackendoff's in (1.48), and the representation used in the current implementation in (1.49). Note that the subcategorisation list in Pinker's entry is missing because he relies on linking theory and that the coindexing is assumed from his examples, and that Jackendoff's lexical entry does not include the subject in the subcategorisation list, following GB.

$$\begin{aligned}
 (1.47) \quad & \left[\begin{array}{c} \text{give} \\ V \\ \left[\text{AFF} (\text{THING}_1, \text{THING}_2, \text{effect} [\text{EVENT:possessional GO} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_3)))) \right] \end{array} \right] ^P \\
 (1.48) \quad & \left[\begin{array}{c} \text{give} \\ V \\ \overline{\text{NP}_j} \text{ } [\text{PP to NP}_k] \\ \left[\begin{array}{c} \text{CAUSE} \left(\left[\text{THING} \right]_i, \left[\text{GO}_{\text{POSS}} \left(\left[\text{THING} \right]_j, \left[\begin{array}{c} \text{FROM}([\text{THING}]_i) \\ \text{TO}([\text{THING}]_k) \end{array} \right] \right) \right) \right) \\ \text{AFF}^+ \left(\left[\text{THING} \right]_i, \right) \end{array} \right] \end{array} \right] ^J
 \end{aligned}$$

$$(1.49) \quad \left[\begin{array}{l} \text{give} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner)} \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to (at (THING}_3)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Next consider the representation of the sentences [1] *The ball rolled* and [2] *The ball rolled down the hill* in each system. These sentences highlight the differences resulting from the differing treatment of adjuncts.

In Pinker's system, the two sentences require two different lexical entries for *roll*. After integration of these lexical entries with the other components of the sentence via linking theory, the representation of [1] would be as in (1.50a), and of [2] as in (1.50b) (Pinker 1989, p. 182).

$$(1.50) \quad \begin{array}{ll} \text{a.} & \left[\text{GO (THING "ball", MANNER "rolling")} \right]^P \\ \text{b.} & \left[\text{GO (THING "ball", down (THING "hill"), MANNER "rolling")} \right]^P \end{array}$$

In Jackendoff's system, only one lexical entry for *roll* is necessary. When used for the analysis of sentence [1] the outcome is as in (1.51a), and when used for sentence [2] with the aid of an adjunct rule the outcome is as in (1.51b) (Jackendoff 1990, p. 224). Note that neither representation indicates the manner in which the MOVING is occurring.

$$(1.51) \quad \begin{array}{ll} \text{a.} & [\text{EVENT MOVE (ball)}]^J \\ \text{b.} & \left[\begin{array}{l} \text{GO (ball}_i, \text{down (at (hill)))} \\ \text{AFF (i,)} \\ \text{[BY [MOVE (i)]]} \end{array} \right]^J \end{array}$$

In the current implementation, both sentences could be parsed using one lexical entry, because ALE could treat the PP as an adjunct, but then the path information added by the PP would not be integrated into the semantics. Thus, it is better to postulate two lexical entries, (1.52a) and (1.52b), which when integrated with the lexical entries for the NP arguments would result in (1.53a) and (1.53b), respectively.

$$(1.52) \quad \begin{array}{ll} \text{a.} & \left[\begin{array}{l} \text{roll} \\ \text{V} \\ \text{NP}_1 \\ \left[\text{GO (THING}_1, \text{PATH, time}_0, \text{rolling_manner)} \right] \end{array} \right] \\ \text{b.} & \left[\begin{array}{l} \text{roll} \\ \text{V} \\ \text{NP}_1, [\text{P}_{\text{form}} \text{NP}_2] \\ \left[\text{GO (THING}_1, \text{pform (at (THING}_2)), \text{time}_0, \text{rolling_manner)} \right] \end{array} \right] \end{array}$$

$$(1.53) \quad \begin{array}{ll} \text{a.} & \left[\text{GO (THING}_1^{\text{ball}}, \text{PATH, time}_0, \text{rolling_manner)} \right] \\ \text{b.} & \left[\text{GO (THING}_1^{\text{ball}}, \text{down (at (THING}_2^{\text{hill}})), \text{time}_0, \text{rolling_manner)} \right] \end{array}$$

Chapter 2

The Dative Alternation

2.1 Description

The dative alternation is the name of a grammatical phenomenon in which ditransitive verbs may appear with varying argument structures. These argument structures are generally referred to as the double object construction and the dative form.

The two objects of ditransitive verbs have traditionally been referred to as the “direct object” and the “indirect object”¹. The direct object generally specifies what the activity reflected in the verb is directed towards. Thus for example, in *He is writing a book*, *a book* specifies what the *writing* activity is directed towards. A direct object is generally needed when the meaning of a verb requires something to give it a focus. The indirect object names the entity for or to whom something is done.

The double object construction is the construction in which the two arguments of a verb appear as two Noun Phrases (NPs) following the verb. The first NP corresponds to the indirect object, while the second NP corresponds to the direct object. The basic word order appears in (2.1).

(2.1) subject + verb + indirect object + direct object

- a. *Kate bought Marie a birthday present.*
- b. *Stuart gave the dog a bone.*

The dative form is the construction in which the indirect object is specified after a preposition. The preposition “to” is generally used if the activity is being done *to* something/someone, and “for” is used if the activity is being done *for* something/someone. The word order appears in (2.2).

(2.2) subject + verb + direct object + *to/for* + indirect object

- a. *Kate bought a birthday present for Marie.*
- b. *Stuart gave a bone to the dog.*

The phenomenon is referred to as an alternation because many verbs which can appear in the double object construction also appear in the dative form, as exemplified in (2.1) and

¹The definitions introduced here and the following usage discussion is based on the Collins Gem guide to English Grammar (Hardie 1990).

(2.2). There do appear to be, however, slight differences in the semantic structure underlying and the pragmatic effect of the two syntactic constructions. The differences in semantic structure will be discussed in both Sections 2.2 and 2.3, while the pragmatic effects will be addressed in Section 2.3.

2.2 Jackendoff's Explanation

Jackendoff's (1990) explanation of the dative alternation hinges on structural differences in the conceptual structures representing the alternate verb forms. He does not attempt to provide a principled account of why some verbs may appear in either of the double object or prepositional forms, while other verbs may only appear in one of the two forms. He does, however, indicate that the use of the double object form is a means of marking the indirect object with a Beneficiary role, and that therefore verbs of possession are likely to alternate.

Verbs which alternate will have two lexical entries, while verbs which do not alternate will only have one entry. Thus the difference between the two types exists purely at the level of the lexicon. The two lexical entries contain differing argument structures which map to different syntactic structures through the mechanism of Jackendoff's linking theory. As he says, "the syntactic reordering of arguments in the dative alternation follows automatically from the alternation in argument structure" (1990, p. 266). Although Jackendoff does identify the core structure of the action tier in each of the forms, no relationship between the entries of the verbs which alternate is postulated and no indication is given of the formal constraints which prevent certain verbs from appearing in one form or the other.

The difference in the two forms involved in the dative alternation is found in the action tier of the conceptual structure. The conceptual structure of the verb in double object form, as for the verb *give* in (2.3), marks the indirect object as a Beneficiary. This is indicated through the use of AFF^+ , whose first argument, the Agent, positively affects the second argument (the indirect object). The conceptual structure of the verb in prepositional form, as in (2.4), on the other hand, treats the direct object (or Theme) as a sort of Patient on which the Agent acts. This Beneficiary/Patient difference is exemplified by the sentences (2.5) (Jackendoff 1990, pp. 135-136), and captures the observation that the inner NP in the double object construction must be the intended possessor of the second NP. The thematic role differences in verb semantics can result in distinct syntactic forms with the aid of appropriate linking rules.

$$(2.3) \quad \left[\begin{array}{c} \text{give} \\ \text{V} \\ \text{NP}_k \text{ NP}_j \\ \left[\begin{array}{c} \text{CAUSE} \left(\begin{array}{c} [\text{THING}]_i, \left[\text{GO}_{\text{POSS}} \left(\begin{array}{c} [\text{THING}]_j, \left[\begin{array}{c} \text{FROM}([\text{THING}]_i) \\ \text{TO}([\text{THING}]_k) \end{array} \right] \right) \end{array} \right) \end{array} \right) \\ \text{AFF}^+ \left(\begin{array}{c} [\text{THING}]_i, [\text{THING}]_k \end{array} \right) \end{array} \right] \end{array} \right]$$

$$(2.4) \quad \left[\begin{array}{c} \text{give} \\ \text{V} \\ \text{NP}_j \text{ [PP to NP}_k] \\ \left[\begin{array}{c} \text{CAUSE} \left(\begin{array}{c} [\text{THING}]_i, \left[\text{GO}_{\text{POSS}} \left(\begin{array}{c} [\text{THING}]_j, \left[\begin{array}{c} \text{FROM}([\text{THING}]_i) \\ \text{TO}([\text{THING}]_k) \end{array} \right] \right) \end{array} \right) \end{array} \right) \\ \text{AFF}^+ \left(\begin{array}{c} [\text{THING}]_i, \end{array} \right) \end{array} \right] \end{array} \right]$$

- (2.5) a. What Harry did for Sam was $\left\{ \begin{array}{l} \text{i. give him a book.} \\ \text{ii. ?give a book to him.} \end{array} \right\}$
- b. i. What Harry did with/*to/*for the books was give every one of them to Sam.
- ii. ?What Harry did with the books was give Sam every one of them.

Jackendoff distinguishes between “true *to*-datives” and “adjunct *to*-datives”. He claims that not all cases in which the double object or *to* prepositional forms occur can be treated alike. For verbs which are “true *to*-datives”, both noun phrases involved in the alternation are true arguments of the verb. For others, the indirect object is better treated as an adjunct. The evidence that the object is not a true argument of the verb comes from the fact that it is always optional and does not seem to contribute a crucial element to the semantics of the verb. Thus, the verbs in (2.6) are true *to*-datives, while the verbs in (2.7) are not (examples from Jackendoff 1990, p. 198). A generalisation over the types of verbs belonging to each class can be observed: true *to*-datives tend to be verbs of possession, while adjunct *to*-datives tend to be spatial verbs. The latter generalisation, however, applies only to verbs which indicate putting something into motion (or “launching” verbs).

- (2.6) a. Adam gave Debbie a book.
Adam served Debbie her dinner.
Adam told Debbie a long story.
Adam paid Debbie \$5.
- b. $\left\{ \begin{array}{l} \text{Adam gave a book} \\ \text{Adam served dinner} \\ \text{Adam told a long story} \\ \text{Adam paid \$5} \end{array} \right\} \left\{ \begin{array}{l} \text{to Debbie.} \\ \text{*out the window.} \\ \text{*down the road.} \\ \text{*into the fire.} \end{array} \right\}$
- c. Adam served/told/paid/*gave Debbie.
- (2.7) a. Sam sent/threw/kicked/hurled/hit Bill the ball.
- b. Sam sent/threw/kicked/hurled/hit the ball to Sandy/out the window/into the park/away.
- c. Sam sent/threw/kicked/hurled/hit the ball.
- d. *Sam sent/kicked/hurled/hit Bill².

Upon a similar analysis, Jackendoff concludes that in all cases of *for*-datives, the indirect object can be treated as an adjunct. In general, the double object form of *for*-datives indicates that the object noun phrase is intended for the benefit of the beneficiary noun phrase, while in the dative form the action as a whole is intended for the benefit of the beneficiary noun phrase³. In order for a *for*-dative verb to appear in the double object form, however, it must be a verb of creation, performance, making available, or preparation. Jackendoff (1990, p. 196) admits that there is no formal method of stating such a constraint within the framework of his theory. For further discussion, see Section 3.3.1.

²In the relevant sense of Bill as an affected entity being moved along a trajectory. These sentences require a prepositional phrase to make the trajectory explicit. *Sam threw Bill* seems to be grammatical, as long as Bill is light enough to be thrown.

³Jackendoff does not, however, show how this implication is captured in the semantic representation of the *for*-beneficiary sentences.

Adjuncts are handled through correspondence rules prescribing the interpretation of argument positions by mapping the syntactic forms into particular conceptual structures. Thus, the *Recipient NP Adjunct Rule* for the double object form of “launching” *to*-datives is represented in (2.8) (from Jackendoff 1990, p. 199).

- (2.8) If V corresponds to [CAUSE_{launch} ([X], [GO ([Y], [PATH])))]
 and NP corresponds to [Z],
 then [_s ... [_{vp} V NP ...] ...] may correspond to
- $$\left[\begin{array}{l} \text{CAUSE}_{\text{launch}}([X], [\text{GO}([Y], [\text{PATH TO}[Z]])]) \\ [\text{FOR } [\text{GO}_{\text{poss}}([Y], \text{TO}[Z])]] \end{array} \right]$$

Similar adjunct rules exist for the other cases for which adjunct rules are appropriate.

2.3 Pinker’s Explanation

Pinker’s account of the dativisability of a particular verb hinges on the semantic structure corresponding to the verb’s meaning. If the structure is compatible with a narrow-range lexical rule, it can alternate; otherwise the structure may be compatible with one of the input or output forms. In this section, I will characterise the broad- and narrow-range rules which Pinker (1989) proposes, and show how they capture the grammatically relevant semantic information relevant to a lexical rule.

As mentioned in Section 1.1.2, the broad-range lexical rule corresponding to the *to*-dative alternation can be characterised as transforming a verb with a semantic structure incorporating “X causes Y to go to Z” into a verb containing a structure “X causes Z to have Y”. This change seems quite subtle, as both entail a change of possession of some object. However, there are clear psychological and pragmatic effects which result from the change. Examples of such effects are found in (2.9) and (2.10) (From Pinker 1989, p. 83). In (2.9a), it is not necessary that the students are actually learning any French, while the implication in (2.9b) is that the teaching is successful and the students are benefitting from the teaching. Likewise, in (2.10a), John is merely a spatial target, while in (2.10b) John was clearly meant to receive the ball.

- (2.9) a. She is teaching French to the students.
 b. She is teaching the students French.
- (2.10) a. I threw the ball to John.
 b. I threw John the ball.

Furthermore, the differences in semantic structure between the double object and prepositional forms are enough to prevent certain verbs from appearing in one of the forms. All dativisable verbs must be capable of denoting prospective possession; thus verbs can alternate only if they signify a transfer of an object that can result in its being possessed.

The *for*-dative alternation maps a verb with a semantic structure incorporating the thematic core “X acts on Y for the benefit of Z” to “X acts on Z by means of acting on Y, and X intends Z to have Y”. The psychological effects observed above with the *to*-dative alternation can also be observed with the *for*-dative alternation. Thus in (2.11b) Mary cannot “win charity the money” because the charity cannot come to possess the money, although if “charity” were construed as the name of a person, both sentences would be acceptable.

- (2.11) a. Mary won the money for charity.
 b. *Mary won charity the money.

2.3.1 Subclass Definitions

In this section, I will introduce the major verb subclasses which Pinker identifies.

verbs of giving (Dativisable)

A giver has some object and causes it to enter into the possession of a recipient.

Examples: give, pass, hand, sell, pay, trade, lend, loan, serve, feed

verbs of communication

- **illocutionary verbs of communication** (Dativisable)

Examples: tell, show, ask, teach, pose, write, spin, read, quote, cite

- **manner of speaking verbs** (Prepositional form only)

Examples: shout, scream, murmur, whisper, shriek, yodel, yell, bellow, grunt, bark

- **verbs specifying an instrument of communication** (Dativisable)

Examples: radio, E-mail, telegraph, wire, fax, telephone

verbs of future having (Dativisable)

X makes some commitment that Y will have or can have Z in the future.

Examples: offer, promise, bequeath, leave, refer, forward, allocate, guarantee, allot, assign, advance, award, reserve, grant

verbs of future not having (Double Object form only)

Examples: cost, spare, envy, begrudge, bet, refuse, ask, save, charge, fine, forgive

verbs of fulfilling/deserving (Prepositional form only)

X transfers Z to Y, where (a) Z is not necessarily possessed by X beforehand; (b) Z is something that Y deserves, needs, or is worthy of; (c) Y's relation to Z has certain properties⁴.

Examples: present, credit, reward, entrust, honor, supply, bestow

verbs of causation of motion

- **verbs of instantaneous imparting of force in some manner causing ballistic motion** (Dativisable)

Examples: throw, toss, flip, slap, kick, poke, fling, blast

- **verbs of continuous imparting of force in some manner causing accompanied motion** (Prepositional form only)

Examples: carry, pull, push, lift, lower, haul

- **verbs of continuous causation of accompanied motion specifying the direction of the motion** (Dativisable)

Examples: bring, take

⁴Note that these verbs can appear in a construction with the prospective possessor as the first object, but this construction then requires the preposition *with* to mark the object transferred, as in *She presented the students with certificates*.

verbs of obtaining

(Dativisable)

X does not initially possess Y, then comes to possess it for Y's benefit so that X can give it over to Y.

Examples: get, buy, find, steal, order, win, earn, grab

verbs of creation

(Dativisable)

In the double object form, expresses the notion of X causing Y to come into existence for the benefit of Z and then causing Z to have Y.

Examples: bake, make, build, cook, sew, knit, xerox

verbs of choosing

(Prepositional form only)

Examples: choose, pick, select, favor, indicate, prefer, designate

2.3.2 Lexical rules

Broad-Range Lexical rules

The broad-range rule for *to*-datives is characterised by (2.12). The broad-range rule for *for*-datives is characterised by (2.13). They each reflect the intuitions about the semantic structure underlying each type, as discussed above.

$$\begin{array}{c}
(2.12) \quad \text{ACT}(\text{THING}_X, \text{THING}_Y, \text{effect}(\text{GO}(\text{THING}_Y, \text{to}(\text{at}(\text{THING}_Z)))) \\
\Downarrow \\
\text{ACT}(\text{THING}_X, \text{THING}_Z, \text{effect}(\text{HAVE}(\text{THING}_Z, \text{THING}_Y)))
\end{array}$$

$$(2.13) \quad \begin{array}{c} \text{ACT (THING}_X, \text{THING}_Y, \text{for/to (HAVE (THING}_Z, \text{THING}_Y))) \\ \updownarrow \\ \text{ACT (THING}_X, \text{THING}_Z, \text{for/to (HAVE (THING}_Z, \text{THING}_Y)), \text{means (ACT (THING}_X, \text{THING}_Y))) \end{array}$$

Narrow-Range Lexical rules

Pinker does not explicitly define the narrow-range lexical rules. Instead, he presents a sample representation of a verb in each narrow subclass which could serve as the input to a lexical rule for that subclass. The parts of this representation which correspond to the structure represented in the broad-range lexical rule would be manipulated in the narrow-range rule precisely as in the broad-range lexical rule.

Pinker (1989, pp. 213-223) discusses variations in semantic structure among elements of a narrow subclass, all of which would have to fit within the input specification of a lexical rule. He also discusses variations in semantic structure which differentiate dativisable subclasses from related nondativisable subclasses. A lexical rule input specification would thus have to be general enough to allow all members of a particular narrow-range subclass to participate in the alternation, yet precise enough to prevent verbs with more "critical" (at least from a syntactic perspective) semantic differences from alternating.

An issue which this discussion raises, then, is whether it is enough to have one narrow-range lexical rule per narrow subclass. Pinker's general position on lexical rules seems to indicate that this would be the case, but his analysis does not support this. This uncertainty about the status of narrow-range lexical rules is due to Pinker's lack of complete formalisation

of the narrow-range lexical rules. Without more precise specification of the rules necessary to account for the narrow-range subclass data and evidence that the rules can account for the data, it is difficult to evaluate the effectiveness of the introduction of lexical rules. If several lexical rules are necessary to capture the variation in each narrow subclass, little is gained over specification of multiple lexical entries for each verb which can alternate. On the other hand, the existence of fewer lexical rules than members of a class does enable prediction of the syntactic alternation properties of newly learned verbs with identical semantic structure as existing verbs. That is, lexical rules capture generalisations about the semantic properties of alternating verbs which are otherwise not immediately obvious.

Pinker's discussion will not be duplicated here. The lexical rules proposed in Section 2.5 incorporate his observations.

2.4 The Lexicon

The lexicon is the place in the HPSG theory in which word-specific information, including phonological information and a word's syntactic and semantic relation to other words is to be located. Each entry must contain information about how the word it represents figures in grammatical rules, which means specifying the syntactic category of the word and associated properties such as the verb form or agreement features.

The SUBCATEGORISATION feature is a critical component for words which characteristically combine with other expressions of specified syntactic categories (the words' *complements*). It is the place in which the complements of the word are fully encoded, as a sequence of categories. In HPSG, the order of (*synsem*) elements on the SUBCAT list corresponds to an *obliqueness hierarchy* of grammatical relations (Pollard and Sag, 1987). This hierarchy is based on the fact that diverse syntactic phenomena obey generalisations that have a hierarchical nature. The hierarchy is shown in (2.14).

$$(2.14) \quad \text{SUBJECT} \Rightarrow \begin{matrix} \text{DIRECT} \\ \text{OBJECT} \end{matrix} \Rightarrow \begin{matrix} \text{INDIRECT} \\ \text{OBJECT} \end{matrix} \Rightarrow \text{OBLIQUES} \Rightarrow \text{GENITIVES} \Rightarrow \begin{matrix} \text{OBJECTS OF} \\ \text{COMPARISON} \end{matrix}$$

The correspondence between the order on the obliqueness hierarchy and surface order is mediated by language-specific grammar rules and principles of linear precedence⁵. In the Gerald Penn and Bob Carpenter implementation of HPSG in ALE, however, no such principles

⁵Discussion of the obliqueness hierarchy and its correspondence to surface order appears in Pollard and Sag 1987. There is, however, a problem with their discussion, as they never define explicitly what they mean by "direct object" and "indirect object". They simply refer to the traditional grammatical relations with these names. Their usage, however, conflicts with the notion of these two types of objects as found in (Hardie 1990). Specifically, they provide the example phrase *give Kim a book* as evidence for the generalisation that less oblique complements precede more oblique complements (pp. 173-174). Given the definitions provided in Section 2.1, it would seem that "Kim" is the indirect object and "a book" the direct object. The hierarchy in (2.14), however, specifies that a direct object is less oblique than an indirect object. This is in direct conflict with the proposed generalisation, as the more oblique complement, the indirect object, precedes the less oblique complement in the example phrase. It is possible that Pollard and Sag simply mean "the object immediately following the verb" when they use the term "direct object". If this is the case, the direct correspondence between relative obliqueness of the direct and indirect objects and their relative surface order is purely a matter of definition and seems rather circular. The terms labelling the objects would no longer have any meaning independent from syntax. The use of the obliqueness hierarchy, grammatical relations, and precedence principles needs to be more explicitly addressed before it can be incorporated into the current version of HPSG theory.

are incorporated⁶, and therefore the order in the SUBCAT list corresponds directly to the surface order of the complements. All lexical entries specified in this document incorporate SUBCAT lists which correspond to surface order.

The main types of verbs to be discussed in the sections that follow are “to”-datives and “for”-datives. The verbs of each type have a common core structure, to which verb-specific variations of elements such as time, manner, additional subordinated structures, THING properties, and semantic field will be attached. For the “to”-datives, the core structure is as found in (2.15), and that of the “for”-datives is in (2.16).

$$\begin{aligned}
 (2.15) \quad & \left[\begin{array}{l} V \\ NP_1, NP_2, \text{ to } NP_3 \\ \left[\begin{array}{l} \text{AFF (THING}_1, \text{ THING}_2, \text{ TIME, MANNER)} \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{SEM_FIELD}} (\text{THING}_2, \text{ to } (\text{at } (\text{THING}_3))), \text{ TIME, MANNER})) \end{array} \right\} \end{array} \right] \end{array} \right] \\
 (2.16) \quad & \left[\begin{array}{l} V \\ NP_1, NP_2, \text{ for } NP_3 \\ \left[\begin{array}{l} \text{AFF (THING}_1, \text{ THING}_2, \text{ TIME, MANNER)} \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{SEM_FIELD}} (\text{THING}_2, \text{ to } (\text{at } (\text{THING}_1))), \text{ TIME, MANNER}), \\ \text{for_to} \\ (\text{HAVE } (\text{THING}_3, \text{ THING}_2, \text{ TIME})) \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{aligned}$$

For the full specification of the lexicon used in this implementation, see Appendix C.

2.5 Lexical Rules

“Lexical Rules provide a mechanism for expressing redundancies in the lexicon, such as the kinds of inflectional morphology used for word classes, derivational morphology as found with suffixes and prefixes, ...” (Carpenter 1993).

Alternations in verbal syntactic form can also be treated as lexical redundancies, when it is observed that the shift from one form to another has consistent semantic consequences. Lexical rules therefore offer a way of capturing these semantic consequences, reflecting intuitions (which can be shown to have linguistic and pragmatic implications) of the semantic meaning conveyed by syntactic form. Furthermore, the existence of multiple syntactic forms for a particular verb also depends on certain aspects of its semantic structure which consistently figure in whether or not a verb will alternate. Lexical rules offer a way of precisely defining what these aspects are.

The current implementation focuses on the process which converts the prepositional dative form to the double object form. Six lexical rules are defined to handle “to”-dative alternation, “for”-dative alternation, and the related alternation “X presents Y with Z” \Leftrightarrow “X presents Z to Y” for verbs of fulfilling. The operation of the lexical rules is governed by the mechanisms of the Attribute Logic Engine (see Carpenter 1993, pp. 43-47). Every lexical entry in the lexicon is checked to see if it matches the input requirements of each lexical rule. This “matching”

⁶This is probably due to the lack of discussion of these principles in the 1994 version of Pollard and Sag’s work on HPSG, on which their grammar is based.

operation actually consists of attempting to unify the lexical entry with the input structure specified by the lexical rule. If they unify, an additional lexical entry is generated according to the structure specified in the output of the lexical rule. Structure sharing between the input and output entries ensures that information in the two entries which should remain the same is introduced into the generated lexical entry.

In addition, lexical rules in ALE allow attachment of definite constraints which are invoked after the lexical rule has matched a lexical entry with the input of the rule. If these definite constraints fail, the application of the lexical rule to the lexical entry fails and no corresponding output structure is generated. The current implementation relies heavily on this procedural attachment to perform set operations on the set of thematic tier elements. These will not be made explicit in the descriptions of the lexical rules to be presented in the following sections. See Appendix D.2 for the actual implemented ALE lexical rules.

2.5.1 “to” Datives

Four of the lexical rules apply particularly to “to” datives. They are approximated by the representations in (2.17)-(2.20)⁷. All of these rules require as input a semantic structure which minimally incorporates the structure common to all “to”-datives (as specified in (2.15) on page 43).

(2.17) Basic “to”-dative Lexical Rule

$$\left[\begin{array}{l} V \\ NP_1, NP_2, \text{ to } NP_3 \\ \left[\begin{array}{l} \text{AFF} (THING_1, THING_2^{\{no \ property, \ simple \ property\}}, PT_IN_TIME_0, MANNER_4) \\ \left\{ \begin{array}{l} effect \\ (GO_{\{possession, \ spatial\}} (THING_2, \text{ to } (at (THING_3))), PT_IN_TIME_5, MANNER_6)), \\ Remaining_Set_Thematic_7 \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\Updownarrow$$

$$\left[\begin{array}{l} V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{l} \text{AFF} (THING_1, THING_3, PT_IN_TIME_0, no_manner) \\ \left\{ \begin{array}{l} effect \\ (HAVE (THING_3, THING_2, PT_IN_TIME_5)), \\ Remaining_Set_Thematic_7 \end{array} \right\} \end{array} \right] \end{array} \right]$$

⁷The representations are “approximated” because many of the features necessary for the HPSG implementation have been left out. Exact specification of these features is not important for the explanation of the lexical rules and their application.

In addition, the representations are not being shown as attribute value matrices for clarity, which means that structure-sharing is indicated by means of coindexing. The relationship between NP_1 and $THING_1$ formally is that they share an `INDEX` value.

(2.18) Communication Class Lexical Rule

$$\begin{array}{c}
 \left[\begin{array}{c} V \\ NP_1, NP_2, \text{ to } NP_3 \\ \left[\begin{array}{c} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{PT_IN_TIME}_0, \text{NO_MANNER}) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{GO}_{\text{communication}} (\text{THING}_2, \text{ to } (\text{at} (\text{THING}_3))), \text{ PT_IN_TIME}_5, \text{MANNER}_4), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right\} \end{array} \right] \end{array} \right] \\
 \Downarrow \\
 \left[\begin{array}{c} V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{c} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{PT_IN_TIME}_0, \text{no_manner}) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{PT_IN_TIME}_5)), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}$$

(2.19) Bring Class Lexical Rule

$$\begin{array}{c}
 \left[\begin{array}{c} V \\ NP_1, NP_2, \text{ to } NP_3 \\ \left[\begin{array}{c} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{continuous}_0, \text{NO_MANNER}) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{GO} (\text{THING}_2, \left[\begin{array}{c} \text{with} (\text{at} (\text{THING}_1)) \\ \text{to} (\text{at} (\text{THING}_3^{\text{deictic}})) \end{array} \right], \text{continuous}_5, \text{MANNER}_4)), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right\} \end{array} \right] \end{array} \right] \\
 \Downarrow \\
 \left[\begin{array}{c} V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{c} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{continuous}_0, \text{no_manner}) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{continuous}_5)), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}$$

(2.20) Fulfilling Class Lexical Rule

$$\begin{array}{c}
 \left[\begin{array}{c} V \\ NP_1, NP_2, \text{ to } NP_3 \\ \left[\begin{array}{c} \text{AFF} (\text{THING}_1, \text{THING}_2^{\text{complex possessional property}}, \text{TIME}_0, \text{MANNER}_4) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{ to } (\text{at} (\text{THING}_3))), \text{ TIME}_5, \text{MANNER}_6), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right\} \end{array} \right] \end{array} \right] \\
 \Downarrow \\
 \left[\begin{array}{c} V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{c} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{TIME}_0, \text{no_manner}) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{TIME}_5)), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}$$

Just as all of the verbs in the “to”-dative form have a common semantic structure, when these verbs are transformed into the double object form they also share a core structure. This structure is shown in (2.21). All of the output structures generated by the rules above incorporate this basic structure.

$$(2.21) \quad \left[\begin{array}{c} V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{c} \text{AFF (THING}_1, \text{THING}_3, \text{TIME, MANNER)} \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{TIME})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Verbs of Giving

The verbs of giving all inherently involve possession transfer. Therefore, these verbs must have the semantic field of their GO effect specified as some type of possession. These verbs are also all instantaneous verbs, in that the occurrences in the action and thematic tiers can be associated with particular points in time, and for each verb there is at most one simple property associated with its direct object. The verbs vary in many other respects, as summarised by Pinker (1989, p. 213):

Pass would specify the semantic field more precisely [than *give*], as “possessional: physical-custody” rather than generic possession (one can *give*, but not *pass*, a car to someone by signing a title transfer agreement; one can *pass*, but perhaps not *give*, an object one doesn’t own). *Hand* would be similar to *pass* with the addition of a MANNER branch specifying the use of the hands. The small subclass that embraces *send*, *mail*, and *ship* would be represented similarly, except in the time-line component of the representation, which would show the ACT event and the GO:possessional event as being linked to distinct event times rather than the single one used in the simple verbs of giving. ...*Sell*, *trade* and *pay* would have a subordinate countertransfer event to which the agent is committed. ...For *pay*, the “money” property would be attached to the [direct object]; for *trade*, it would be absent.

Pinker suggests that a special narrow-range lexical rule would be necessary for the *sell*, *trade* and *pay* subclass, in addition to a more general narrow-range rule for the other verbs in the *giving* class. In the current implementation, all of the varied semantic representations can be accommodated by the Basic “to”-dative lexical rule in (2.17). This rule incorporates the elements which all verbs of giving share, in addition to the core “to”-dative structure. In particular, it restricts the STATES and EVENTS to be instantaneous and the direct object to having at most one simple property, and allows the subordinated GO event to be one of possession, but does not limit the type of MANNER specification or the presence of additional subordinated branches. The latter feature of the rule allows it to be general enough to accommodate the within-narrow-subclass variances.

An example of the use of this lexical rule is its application to the lexical entry for *sell*. The lexical entry for *sell* found in the lexicon is (2.22a). After this entry is used as input to the lexical rule, the generated entry is (2.22b). The method by which the entry is generated is as follows. The lexical entry for *sell* unifies with the input structure specified by the Basic “to”-dative lexical rule, in the process instantiating *Remaining_Set_Thematic* to the set

containing only the *obligates* subordinating function and its argument. The output structure is then created based on the specification in the lexical rule. This handles the shuffling of the SUBCAT list and the arguments in the action tier. It also replaces the EVENT subordinated by the *effect* relation in the original lexical entry with the STATE required by the semantics of the double object form. Through the use of structure sharing in the lexical rule, the NPs in the output structure will carry the same INDEX values as the corresponding NPs in the input structure, which will in turn be appropriately shared with the THING elements in both the input and output semantic structures. Structure sharing also accounts for the use of the same TIME values in both structures, and combined with the set operations attached to the lexical rule, the presence of *Remaining_Set_Thematic* in the output structure.

$$\begin{array}{lcl}
 (2.22) & \text{a.} & \left[\begin{array}{l} \text{sell} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner})), \\ \text{obligates} \end{array} \right\} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_3, \text{THING}_4^{\text{money}}, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_4, \text{to}(\text{at}(\text{THING}_1)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right] \end{array} \right] \\
 & \text{b.} & \left[\begin{array}{l} \text{sell} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_0), \\ \text{obligates} \end{array} \right\} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_3, \text{THING}_4^{\text{money}}, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_4, \text{to}(\text{at}(\text{THING}_1)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right] \end{array} \right]
 \end{array}$$

For each of the verbs of giving, a lexical entry for the double object form will be generated based on the “to”-dative form, by the same method as the entry for the double object form of *sell*. Examples of the lexical entries for these verbs can be found in Appendix C.1. The generated forms can be seen in Appendix C.2.

None of the other lexical rules will apply to these verbs. The Communication Class lexical rule requires the semantic field of the GO event to be **communication**, which these verbs do not satisfy; the Bring Class lexical rule requires the time events to be continuous, while these verbs specify instantaneous events; the Fulfilling Class lexical rule requires the direct object to have a complex property, while none of these verbs incorporate complex properties in their semantic structure.

Verbs of Communication

This class consists of three subclasses: the dativisable class of verbs involving the communication of ideas, the nondativisable class of verbs of manner of speaking, and the dativisable class of verbs specifying an instrument of communication. Pinker’s summary (1989, pp. 214-215) of these classes appears below.

[Verbs of communication] involve a subfield of possession involving the communication of ideas. In addition, verbs like *tell*, *ask*, and *write* differ by virtue of specifying messages with different illocutionary force, differentiated with respect to an intended effect on a hearer. That is, the message is such that a hearer is supposed to come to know it (*tell*), learn it (*teach*), answer it (*ask*, *pose*), ...[t]hat would result in a semantic structure like that for *tell*, in which the tellable argument is constrained by the ‘for/to’ property substructure to be something that a listener is supposed to be able to know ... One can easily represent verbs of instrument of communication using a variation [in which the] PROPERTY specification for the [direct object] would be omitted, and a ‘means’ substructure would be added in which the agent would ACT on a THING specified as a quoted constant ... [For verbs of manner of speaking] the field of the root event must be “physical”, because that is the field in which the MANNER must be interpreted and the verb specifies a manner. The theme is specified to be sound so as to rule out **Bob shouted some spit to John*. Other verbs in the class would have identical representations except for the quoted manner specification.

Thus, verbs in the first subclass have identical structure, differing only in the STATE specified in the complex *for-to* property associated with the direct object. Verbs in the instrument of communication class differ as indicated above, and verbs of manner of speaking are quite different from the other two classes, but verbs within the class differ only in the MANNER they specify.

Two examples of the first subclass, *tell* and *ask*, two examples from the second subclass, *shout* and *whisper*, and one example from the third, *radio* have been handled in the implementation. *Ask* is shown in (2.23a), its generated form in (2.23b), *shout* in (2.24), and both forms of *radio* in (2.25).

The generated forms of the verbs of communication and instrument of communication are produced by the Communication Class lexical rule in (2.18), which specifies the semantic field of the GO event explicitly as **communication**, but accommodates the differences in the PROPERTY associated with the direct object and the additional *means* substructure of the instrument class verbs. The former is handled by not placing any restrictions on the value of a PROPERTY field (thus one need not even exist), and the latter by structure sharing *Remaining_Set_Thematic* between the input and output forms. Neither the Basic “to”-dative lexical rule nor the Fulfilling Class lexical rule apply to these verbs, due to the incompatibility of the semantic fields of the subordinated GO event of these verbs with that required by the rules. The Bring Class lexical rule doesn’t apply because it requires the EVENTS to be continuous, which is not the case for these verbs.

No forms for the verbs of manner of speaking are generated. These verbs do not match any of the lexical rules due to the **physical** field in which the action tier must be interpreted, combined with the instantaneous nature of the EVENTS and STATES.

- (2.23) a.
$$\left[\begin{array}{l} \text{ask} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2 \textit{for_to}(\text{BE}_{\text{epistemic}}(\text{THING}_2, \textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{communication}}(\text{THING}_2, \textit{to}(\textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner}), \\ \textit{for_to} \\ (\text{GO}_{\text{communication}}(\text{THING}_4^{\textit{answer}}, \textit{to}(\textit{at}(\text{THING}_1))), \text{time}_1, \text{no_manner}) \end{array} \right\} \end{array} \right] \end{array} \right]$$
- b.
$$\left[\begin{array}{l} \text{ask} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2 \textit{for_to}(\text{BE}_{\text{epistemic}}(\text{THING}_2, \textit{at}(\text{THING}_3))), \text{time}_0)), \\ \textit{for_to} \\ (\text{GO}_{\text{communication}}(\text{THING}_4^{\textit{answer}}, \textit{to}(\textit{at}(\text{THING}_1))), \text{time}_1, \text{no_manner}) \end{array} \right\} \end{array} \right] \end{array} \right]$$
- (2.24) a.
$$\left[\begin{array}{l} \text{shout} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF}_{\text{physical}} (\text{THING}_1, \text{THING}_2^{\textit{sound}}, \text{time}_0, \text{shouting_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{perceptual}}(\text{THING}_2, \textit{to}(\textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner}) \end{array} \right\} \end{array} \right] \end{array} \right]$$
- b. No double object form generated for *shout*.
- (2.25) a.
$$\left[\begin{array}{l} \text{radio} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{communication}}(\text{THING}_2, \textit{to}(\textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner}), \\ \textit{means} \\ (\text{AFF}(\text{THING}_1, \text{THING}^{\textit{radio}}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$
- b.
$$\left[\begin{array}{l} \text{radio} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_0)), \\ \textit{means} \\ (\text{AFF}(\text{THING}_1, \text{THING}^{\textit{radio}}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Verbs of Future Having

This is a dativisable class of verbs for which

the main event is not an act of giving but an act of commitment, with the possession transfer an “effect” of the commitment bound to a different event on the time-line. The act of commitment can be contractual (*bequeath*, *guarantee*, *reserve*, *assign*, *allot*, *leave*) or verbal (*refer*, *recommend*, *offer*), but in all cases the act by its nature involves a designated future possessor. ...[T]hese verb-specific pieces of information are subordinated as *means* substructures, capturing the intuition that the main event is the act of commitment that has as its effect a future possession transfer. Offering, assigning, recommending, bequeathing, and so on, differ primarily in having different means of bringing about that future event. (Pinker, 1989, p. 216)

These verbs are therefore identifiable by the distinct points in time at which the action tier and the EVENTS in the thematic tier occur. Their alternation is handled by the Basic “to”-dative lexical rule, as their structure is not compatible with any of the others. The representation for *bequeath* is in (2.26a), and the generated lexical entry is in (2.26b). Other verbs in this class differ in the nature of the EVENT that is subordinated by the *means* relation.

$$\begin{array}{ll}
 (2.26) & \text{a.} \quad \left[\begin{array}{l} \text{bequeath} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_2, \text{no_manner})), \\ \text{means} \\ (\text{BE}(\text{THING}^{\text{will}}, \text{THING}^{\text{existence}}, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array} \right] \\
 & \text{b.} \quad \left[\begin{array}{l} \text{bequeath} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_2)), \\ \text{means} \\ (\text{BE}(\text{THING}^{\text{will}}, \text{THING}^{\text{existence}}, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}$$

Verbs of Future Not Having

These are verbs that appear only in the double object form. They were not treated in this implementation, as all of the lexical rules involve transforming verbs in the dative form to the double object form and not vice versa. Treating these verbs would simply entail having one lexical entry for each of the verbs specifying that the verb appears in the double object form and structure sharing syntactic and semantic arguments as appropriate.

Verbs of Fulfilling/Deserving

The verbs of fulfilling can appear in several different forms. A certain subset, such as *{present, furnish, provide, supply, entrust, credit}* alternate between the forms “*X presented Z with Y*” and “*X presented Y to Z*”. For some speakers, these also alternate to the double object form “*X presented Z Y*”. The verbs *reward* and *honor*, however, can either appear in the *with* form or as transitives, i.e. “*X rewarded Y*”, but not in either of the “to”-dative forms or the double object form.

A lexical rule was developed to handle the alternation between the “with-prepositional” and “to”-dative forms. It is shown in (2.27).

(2.27) Fulfilling Class “with preposition” Lexical Rule

$$\left[\begin{array}{l} V \\ NP_1, NP_3, \text{ with } NP_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{TIME}_0, \text{MANNER}_4) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{TIME}_0, \text{MANNER}_6)), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right\} \end{array} \right] \end{array} \right] \\ \Updownarrow \\ \left[\begin{array}{l} V \\ NP_1, NP_2, \text{ to } NP_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{TIME}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{TIME}_0, \text{MANNER}_6)), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right\} \end{array} \right] \end{array} \right]$$

This rule applies to the verb *present*, shown in (2.28a), to produce (2.28b). The rule applies to the lexical entry of the *present*-type verbs but not to that of the *reward*-type verbs because the former match the input structure required by the lexical rule, while the latter do not because rather than the required $\text{GO}_{\text{possession}}$ event, they have a *HAVE* event subordinated by the *effect* relation. Thus the “to”-dative form is not generated for *reward*-type verbs.

The Fulfilling Class lexical rule was introduced to handle the alternation from the “to”-dative to the double object form for those verbs which alternate for some speakers. For those speakers which do not allow the double object form for those verbs, the lexical rule is simply not a part of their lexical system. The rule transforms the lexical entry for *present*-type verbs in “to”-dative form, as in (2.28b), to the double object form, shown in (2.28c). The input required by this lexical rule matches exactly the output generated by the Fulfilling Class “with preposition” lexical rule, and since all “to”-dative forms of the *present*-type verbs must have generated by that lexical rule, all verbs of that type will alternate with the double object form as well. Similarly, the rule obviously cannot apply to *reward*-type verbs, as no “to”-dative form will have been generated from the “with preposition” lexical rule.

$$\begin{aligned}
 (2.28) \quad & \text{a.} \quad \left[\begin{array}{l} \text{present} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{ with NP}_2 \\ \left[\text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \right. \\ \left. \left\{ \begin{array}{l} \text{effect} \\ \left(\begin{array}{l} \text{GO}_{\text{possession}}(\text{THING}_2 \\ \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner}) \end{array} \right) \end{array} \right\} \\ \left. \left(\begin{array}{l} \text{for_to} \left(\begin{array}{l} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ (\text{EVENT}(\text{THING}_3)) \end{array} \right) \end{array} \right) \right\} \right] \end{array} \right] \\
 & \text{b.} \quad \left[\begin{array}{l} \text{present} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{ to NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \\ \left[\begin{array}{l} \text{for_to} \left(\begin{array}{l} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ (\text{EVENT}(\text{THING}_3)) \end{array} \right) \end{array} \right] \end{array} \right] \\
 & \text{c.} \quad \left[\begin{array}{l} \text{present} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{ NP}_2 \\ \left[\text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \right. \\ \left. \left\{ \begin{array}{l} \text{effect} \\ \left(\begin{array}{l} \text{for_to} \left(\begin{array}{l} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ (\text{EVENT}(\text{THING}_3)) \end{array} \right) \end{array} \right) \end{array} \right\} \\ \left. \left(\begin{array}{l} \text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_0) \end{array} \right) \right\} \right] \end{array} \right] \\
 (2.29) \quad & \text{a.} \quad \left[\begin{array}{l} \text{reward} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{ with NP}_2 \\ \left[\text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \right. \\ \left. \left\{ \begin{array}{l} \text{effect} \\ \left(\begin{array}{l} \text{HAVE}(\text{THING}_3, \text{time}_0) \left\{ \begin{array}{l} \text{for_to} \left(\begin{array}{l} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ (\text{EVENT}(\text{THING}_3)) \end{array} \right), \text{reward} \end{array} \right\} \end{array} \right) \end{array} \right\} \right] \end{array} \right]
 \end{aligned}$$

$$\begin{array}{l}
 \text{b.} \quad \left[\begin{array}{l} \text{reward} \\ \text{V} \\ \text{NP}_1, \text{NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2^{\text{reward}}, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}$$

Verbs of Causation of Motion

The three subclasses of this class are dativisable verbs of instantaneous imparting of force in some manner causing ballistic motion (*throw*-type verbs), nondativisable verbs of continuous imparting of force in some manner causing accompanied motion (*pull*-type verbs), and dativisable verbs of continuous causation of accompanied motion specifying the direction of the motion (*bring*-type verbs). Pinker summarises (1989, pp. 218-219),

The crucial difference is an interaction between aspectual and force-dynamic components of the event: for *throw* verbs, the causing act is an instantaneous even preceding the motion of the object; for *pull* verbs, it is a continuous process that is temporally coextensive with the motion of the object. ... [*Bring*-type verbs] differ from the *pull* verbs in not specifying a manner, in specifying deictic information concerning the path, and also in implying that the agent moves—one can *pull* a box either while staying in one place (using a rope) or by moving with the object, but one can't *take* or *bring* a box anywhere while seated on a rock.

The lexical entries (original and, if possible, generated) for *throw*, *pull*, and *bring* appear in (2.30)-(2.32).

$$\begin{array}{l}
 (2.30) \quad \text{a.} \quad \left[\begin{array}{l} \text{throw} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{throwing_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}$$

$$\begin{array}{l}
 \text{b.} \quad \left[\begin{array}{l} \text{throw} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{throwing_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}$$

$$\begin{array}{l}
 (2.31) \quad \text{a.} \quad \left[\begin{array}{l} \text{pull} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{continuous}, \text{pulling_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{continuous}, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}$$

b. No double object form generated for *pull*.

$$(2.32) \quad \begin{array}{l} \text{a.} \quad \left[\begin{array}{l} \text{bring} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF}_{\text{physical}} (\text{THING}_1, \text{THING}_2, \text{continuous}, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{spatial}} (\text{THING}_2, \left[\begin{array}{l} \text{with (at (THING}_1)) \\ \text{to (at (THING}_3^{\text{here}))} \end{array} \right], \text{continuous}, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right] \\ \\ \text{b.} \quad \left[\begin{array}{l} \text{bring} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3^{\text{here}}, \text{continuous}, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{continuous})) \end{array} \right\} \end{array} \right] \end{array} \right] \end{array}
 \end{array}$$

The *throw*-type verbs alternate via the Basic “to”-dative lexical rule, as it allows for the specification of MANNER, the **spatial** semantic field on the subordinated GO event, and the different points in time associated with the action and thematic tiers. The *bring*-type verbs alternate via the Bring Class lexical Rule, whose input requirements mimic the structure of the lexical entry for *bring*, allowing variations only in the properties attached to the arguments, the type of deictic reference in the indirect object, manner specification in the subordinated GO structure, and additional subordinated structures.

The *pull*-type verbs are prevented from alternating because they do not match any of the lexical rules. They do not match the Basic, Communication Class, and Fulfilling Class lexical rules because they specify continuous as opposed to instantaneous events. They do not match the Bring Class lexical rule because they specify a manner in the action tier, which is not allowed by that rule.

2.5.2 “for” Datives

The single remaining lexical rule suffices to handle all of the “for”-dative alternations. This rule is defined as in (2.33), with the additional criteria that it will not add a *means* subordinated structure if one already exists in *Remaining_Set_Thematic* in the input structure. No core structure for the double object form of the “for”-datives will be shown separately, as it is clear from the output of this lexical rule.

$$(2.33) \quad \text{“for”-dative Lexical Rule}$$

$$\left[\begin{array}{l} \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{TIME}_0, \text{MANNER}_4) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO} (\text{THING}_2, \text{PATH}_8)), \text{TIME}_5, \text{MANNER}_6) \end{array} \right\} \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_9)), \\ \text{Remaining_Set_Thematic}_7 \end{array} \right] \end{array} \right]$$

\Updownarrow

$$\left[\begin{array}{l} V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{l} AFF (THING_1, THING_3, TIME_0, MANNER_4) \\ \left\{ \begin{array}{l} effect \\ (GO (THING_2, PATH_8)), TIME_5, MANNER_6)), \\ for_to \\ (HAVE (THING_3, THING_2, TIME_9)), \\ means \\ (AFF (THING_1, THING_2, TIME_5, no_manner)), \\ Remaining_Set_Thematic_7 \end{array} \right\} \end{array} \right] \end{array} \right]$$

Verbs of Obtaining

All representations for the verbs of obtaining incorporate a structure like that of the verb *get*. Variations on this structure are summarised below.

Buy specifies a caused obligation of a countertransfer of an object with “money” properties to a third party; *grab* specifies a physical manner; *win*, *earn* and *order* involve means; *find* and *steal* specify properties of the obtained object... (Pinker 1989, p. 222)

The base and generated lexical entries for *get* are represented as in (2.34); those for *buy* are in (2.35). *Buy* has been chosen as an example due to the complexity of its semantic representation. The lexical rule for “for”-datives handles the transformations of the entries because all verbs of obtaining have an action tier and a thematic tier consisting of at least *effect* and *for_to* substructures. Since the lexical rule is not particular about the semantic field of the GO event subordinated by *effect* (which is **possession** for all of the verbs of obtaining), all of these entries match precisely its required input, causing them to alternate with the double object form, as they should.

$$(2.34) \quad \begin{array}{ll} \text{a.} & \left[\begin{array}{l} get \\ V \\ NP_1, NP_2, for NP_3 \\ \left[\begin{array}{l} AFF (THING_1, THING_2, time_0, no_manner) \\ \left\{ \begin{array}{l} effect \\ (GO_{possession} (THING_2, to (at (THING_1)), time_0, no_manner)), \\ for_to \\ (HAVE (THING_3, THING_2, time_1)) \end{array} \right\} \end{array} \right] \end{array} \right] \\ \text{b.} & \left[\begin{array}{l} get \\ V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{l} AFF (THING_1, THING_3, time_0, no_manner) \\ \left\{ \begin{array}{l} effect \\ (GO_{possession} (THING_2, to (at (THING_1)), time_0)) \\ for_to \\ (HAVE (THING_3, THING_2, time_1)) \\ means \\ (AFF (THING_1, THING_2, time_0, no_manner)) \end{array} \right\} \end{array} \right] \end{array} \right] \end{array}$$

$$\begin{array}{ll}
 (2.35) & \text{a.} \quad \left[\begin{array}{l} \text{buy} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_4^{\text{money}}, \text{time}_2, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_4, \text{to} (\text{at} (\text{THING})), \text{time}_2, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right] \\
 & \text{b.} \quad \left[\begin{array}{l} \text{buy} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \text{means} \\ (\text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner})), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_4^{\text{money}}, \text{time}_2, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_4, \text{to} (\text{at} (\text{THING})), \text{time}_2, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]
 \end{array}
 \end{array}$$

Verbs of Creation

As with verbs of obtaining, there is a basic structure which is incorporated into the representations for verbs of creation. This is the structure of the verb *make*, shown in (2.36). Other verbs of creation have structures which are only slightly variant, with added means subordinated structures and properties. These verbs have essentially the same structure as the verbs of obtaining, the main difference being that they specify the semantic field of the subordinated GO event to be existential (something “comes into existence”), but again the lexical rule’s lack of specificity about the semantic field of this GO event enables all of these entries to match precisely its required input, and therefore to alternate as they should.

The verb *bake*, represented in (2.37), shows that when a verb’s semantic structure already contains a *means* subordinated structure, no new *means* structure is added, but the remainder of the alternation processes do occur.

- (2.36) a.
$$\left[\begin{array}{l} \text{make} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to (at (existence))), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \end{array} \right] \end{array} \right]$$
- b.
$$\left[\begin{array}{l} \text{make} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to (at (existence))), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \\ \text{means} \\ (\text{AFF (THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner})) \end{array} \right] \end{array} \right]$$
- (2.37) a.
$$\left[\begin{array}{l} \text{bake} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to (at (existence))), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ \text{AFF (THING}_1, \text{THING}^{\text{oven}}, \text{time}_0, \text{no_manner}) \end{array} \right\} \end{array} \right] \end{array} \right]$$
- b.
$$\left[\begin{array}{l} \text{bake} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to (at (existence))), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \\ \text{means} \\ (\text{AFF (THING}_1, \text{THING}^{\text{oven}}, \text{time}_0, \text{no_manner})) \end{array} \right] \end{array} \right]$$

Verbs of Choosing

Pinker does not explicitly define the structure of the nondativisable verbs of choosing, indicating only (Pinker 1989, pp. 114, 223) that they may be “compatible with a goal of transferring [a] chosen object to another party” although there is an element of meaning involving obtaining the object which is missing, and that they are represented similarly to verbs of touching, giving them a semantic structure significantly different from other “for”-datives to prevent them from participating in the dative alternation.

In this implementation, the verbs of choosing have been represented as in (2.38) and (2.39), wherein the “missing” element of meaning is reflected by the absence of a GO event subordinated by the *effect* relation. This missing element of meaning in turn prevents these lexical entries from matching the input required by the “for”-dative lexical rule, and thus from being transformed into the double object form.

$$(2.38) \left[\begin{array}{l} \{ \text{choose, select, pick} \} \\ V \\ NP_1, NP_2, \text{ for } NP_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2^{\text{preferred}}, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{cause} \\ \left(\begin{array}{l} \text{AFF}^+ (\text{THING}_2, \text{THING}_1, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{cause} \\ (\text{AFF}^+ (\text{THING}_2, \text{THING}_3, \text{time}_2, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} , \end{array} \right] \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_2)) \end{array} \right] \end{array} \right]$$

$$(2.39) \left[\begin{array}{l} \{ \text{favor, prefer} \} \\ V \\ NP_1, NP_2, \text{ for } NP_3 \\ \left[\begin{array}{l} \text{AFF}^+ (\text{THING}_2^{\text{preferred}}, \text{THING}_1, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{cause} \\ (\text{AFF}^+ (\text{THING}_2, \text{THING}_3, \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Some discussion is warranted about the semantic structures designated above. *Choosing* (or *picking* or *selecting*) an object for someone involves deciding which object is wanted from a range of things or people, i.e. identifying the preferred object, with the intention that the person should have the preferred object. Of course, the object is probably “preferred” relative to the person for whom it is being identified. That is, it is a decision based on what the person for whom the item is being selected would prefer. The verbs are represented as above to suggest that if *X chooses Y for Z*, then “X affects Y, because Y is preferred, in order for Z to have Y. Y is preferred since it affected X positively, because it is expected to affect Z positively.” *Favor* and *prefer* are similar, but they primarily express a mental state of preference, specifically, “Y is preferred since it affects X positively, because it is expected to affect Z positively”. Thus, the semantic structure of these two verbs do not represent X affecting Y as the primary component of the action tier and do not include the subordinated *for_to* structure in their representation since they do not necessarily intend Z to have Y.

Pinker also includes *indicate* and *designate* as examples of choosing-type “for”-datives. It seems, however, that any interpretation for such sentences as “John designates the message for Mary” and “Bill indicates the bike for Howard” is unnatural. These have not been included in the implementation.

Chapter 3

Discussion

3.1 Lexical Rules

3.1.1 The Number of Lexical Rules

Pinker (1989) proposes that one lexical rule is necessary to handle the alternation of each narrow subclass. For the dative alternation, this means a minimum of eight lexical rules for the “to”-datives and two for the “for”-datives. More might actually be necessary due to Pinker’s basic approach to lexical rules, in which each rule specifies an input structure defining the verbs to which it can apply. From his discussion, it is difficult to discern how the lexical rules would be formulated to accommodate variations in the semantic structures within a narrow class or what the process is for determining whether a structure matches the input structure of a rule. Furthermore, when Pinker identifies subclasses *within* narrow classes which differ in some significant way from the other verbs in the class but still do not behave differently syntactically, it is unclear whether he has in mind individual lexical rules to handle the subclass as distinct from other verbs in the class, or whether it is somehow possible to accommodate all the verbs in the narrow class by one lexical rule. Since Pinker does not explicitly state all of his lexical rules or explain how each applies to a relevant set of verbs, it cannot be determined precisely how restrictive each of his lexical rule definitions would be, or how many are necessary.

By taking advantage of the mechanisms of HPSG, in particular type unification, and generalising over the similarities and differences between alternating and non-alternating classes of verbs, it is possible to dramatically reduce the number of lexical rules necessary to model the dative alternation. Certain features, in particular those features which are not semantic criteria relevant to the syntactic alternation, of the input structure requirements of a lexical rule are left unspecified in the lexical rule. This genericness allows verb structures to have varying values for these features (as long as the value it has is still a subtype of the generic type which the feature value must specify) and still match the requirements for alternation via the lexical rule. For example, if the value of the feature `sem_field` is left unspecified for some STATE or EVENT in the lexical rule, then any semantic field (`possession`, `physical`, etc.) would unify with the general type `sem_field`. Verbs which vary in the semantic field they specify would all alternate, provided they matched the other, more specific, semantic structural elements included in the input requirements. The application of such mechanisms was shown in the previous section.

It was possible to reduce the number of lexical rules Pinker required to handle the dative

alternation by defining lexical rules more precisely than Pinker's broad-range rule, but less precisely than any of his narrow-range rules. The key to this was identification of the crucial elements in differentiating alternating and nonalternating verbs. These appear to be the property associated with the direct object, the semantic field of the subordinated EVENT common to all verbs in the "to"- and "for"-dative forms, and the temporal nature of the occurrences in the action and thematic tiers. Consequently, the input requirements of the lexical rules for the "to"-dative form vary only in these three elements, and verbs either match or don't match the input structures due to the values of these elements.

3.1.2 Violations of Lexical Rules

As mentioned in Section 2.5 in the discussion of verbs of fulfilling, there can be differences among speakers as to the grammatical acceptability of certain constructions. Speakers will sometimes violate constraints such as the morphological constraint, as in (3.1), or use verbs in the double object construction when they do not generally alternate, as in (3.2) (examples from Pinker 1989, pp. 155-156).

- (3.1) a. Sue donated them a bunch of computers.
- b. I returned her the books.
- (3.2) a. Can you reach me that book?
- b. I put you out a big piece [of pie].

Upon analysis of these examples and similar ones, it seems that the violations occur most frequently when the indirect object is a pronoun. Thus although (3.1b) sounds somewhat odd, it does not sound completely wrong, in contrast to e.g. *I returned the store the books* or *I returned Mary the books*. It is unclear why these types of violations would occur. It may have something to do with the fact that pronouns are used to refer to information in the discourse which is known, and that speakers tend to place such "old" or "given" information as close to the front of a sentence as possible, while "new" information is generally introduced later in the sentence¹. These types of violations deserve further investigation, particularly with regard to the semantic implications of using the double object form with a pronominal object for verbs which typically don't alternate.

A different type of apparent lexical rule violation occurs when certain verbs which normally can appear in both the dative and double object forms are ungrammatical in one of the two forms with particular arguments. The verb *give* is the most obvious example of this. The alternation in (3.3) is clearly grammatical, while those in (3.4) and (3.5) are not.

- (3.3) a. John gave his daughter a book.
- b. John gave a book to his daughter.
- (3.4) a. John gave his daughter a bath.
- b. *John gave a bath to his daughter.
- (3.5) a. John gave his daughter a kiss.
- b. *?John gave a kiss to his daughter.

¹For example, it would be more natural to say *There is a cat in the garden* than *A cat is in the garden*, while *The cat is in the garden*, in which the cat being referred to is already known, is completely natural. (Example from Patrick Sturt, personal communication)

In (3.4), the interpretation depends on a particular sense of the word *bath*. There are two senses of this word, one of them being the physical bathtub and the other being the washing that is done inside the bathtub. The sentence (3.4b) is perfectly grammatical if *bath* is interpreted as a bathtub (for example, if John’s daughter were setting up a house, he could be contributing to the contents of the bathroom). The problem which arises when attempting to interpret the dative form with the non-object sense of *bath*, or the dative form in (3.5b), is that John cannot **possess** that type of bath or a kiss in any obvious way, and therefore there can be no transfer of possession of it. This transfer of possession is an explicit component of the semantic structure of the dative form (as expressed by the subordinated GO_{possession} event in the core structure shared by all “to”-datives, shown in (2.15) on page 43). In contrast, the double object form does not indicate any kind of transfer of possession, but requires the indirect object to HAVE the direct object². Furthermore, the double object form treats the indirect object as the affected entity, thus capturing the sense of interaction we have between “John” and “his daughter” in examples (3.4) and (3.5). In the dative form, on the other hand, the direct object is the affected entity. It is difficult to understand how “John” could affect “a bath” or “a kiss”. In sum, the semantics of the dative form are incompatible with the type of entity expressed by the indirect object in (3.4) and (3.5). Treatment of this problem would likely involve enforcement of a selectional restriction indicating that the indirect object must be compatible with a notion of possession and possession transfer in the dative form. Possibly it would be enough to require the indirect object to be a concrete entity; this claim would, however, need to be verified by investigation of uses of the dative form.

3.2 The Dative Alternation

3.2.1 The Morphophonological Constraint on the Dative

It can be observed that verbs which have Latinate roots often do not dativise, despite an underlying semantic structure analogous to that of Germanic dativisable verbs. Consider, for example, the verb *donate*. Its semantics are almost identical to that of *give*, perhaps with the additional implications that the agent doing the *donating* “must have publicly charitable motives” (Pinker, p. 123) and the recipient must be an organisation or a representative of an organisation, yet it does not dativise.

Obviously children’s ability to learn this difference cannot be explained through etymological differences in the verbs, as children cannot be aware of language history. Pinker suggests that the historical differences manifest themselves as morphological differences which children **can** pick up on as they acquire verb argument structure. Latinate verbs tend to be polysyllabic, while Germanic verbs are generally monosyllabic or polysyllabic with stress on the first syllable. Latinate verbs which have similar stress patterns as the Germanic verbs tend to dativise, as shown by the verbs with contrasting stress in (3.6) (from Pinker 1989, p. 46). This morphophonological constraint on dativisation has been supported by studies with invented words (see Pinker 1989, p. 53), but does not appear to apply to all verb classes³.

²While HAVING a bath seems to involve a slightly different sense of HAVE than HAVING a book, this difference has not been treated in this implementation. The distinction is also apparent in the phrase *have a drink*. This use of HAVE seems to be a more abstract notion of possession, but it is not entirely obvious how to define or label it, or whether it affects the linguistic phenomena being treated here in any way.

³Specifically, the dativisable subclasses of verbs of future having and verbs of instrument of communication are insensitive to the morphological constraint

- (3.6) a. Promise/Offer/Recommend/Describe anything to her, but give her Arpège.
 b. Promise/Offer/*Recommend/*Describe her anything, but give her Arpège.

Pinker (1989, pp. 121-122) explains the interaction of the morphophonological constraint with the dative alternation as follows:

In general, lexical rules can effect simultaneous changes in semantics, argument structure syntax, and morphology. The morphological change is seen in English only in the passive ...[b]ut in several other languages, these alternations involve specific morphological changes ...The sensitivity of the English dative rule to morphological class could then be a consequence of two assumptions:

1. Morphological rules can be selective in their application to different morphological classes.
2. Rules that alter argument structures count as morphological rules, even if they do not effect an overt morphological change.

Thus the English dative rule, though it has no overt morphological operations, is formally a kind of rule that can have morphological operations, and therefore it can be sensitive to salient morphological subclasses in the vocabulary of the language.

The morphophonological constraint on the dative alternation has not been handled in this implementation. It is not entirely clear how to handle the selective nature of the constraint given the existing implementation. If the constraint applied to all verb subclasses, it would be fairly straightforward to handle with the addition of an additional feature to the head features of a verb identifying the morphological status of the verb (i.e. mono- vs. polysyllabic and its stress pattern). The feature could then be required to be either monosyllabic or to have stress on the first syllable in order for the lexical entry of the verb to satisfy the input requirements of a lexical rule. Handling of the selective nature of the constraint would require the addition of two lexical rules – one for each of the two classes to which it does not apply, with fully defined input structures to match the precise semantic structure of the verbs in the two subclasses, but without the morphological requirement.

3.2.2 Directionality

Jackendoff refers to the dative alternation as “a syntactic alternation that appears to carry secondary semantic consequences” (Jackendoff 1990, p. 286). Elsewhere he says, “the syntactic reordering of arguments in the dative alternation follows automatically from the alternation in argument structure” (Jackendoff 1990, p. 267). This calls into question the directionality of alternation phenomena. Particularly, does the syntactic change affect the semantics, or is the alternation in syntax driven by a change in the semantic structure to be expressed? The perspective adopted in the current work remains neutral on this issue. It simply makes available a lexical entry for each of the different syntactic and corresponding semantic forms, indicating that a particular semantic structure must be associated with a particular syntactic form but not specifying anything about the basis for this association.

3.3 Implementational Issues

3.3.1 Arguments vs. Adjuncts

Following Pinker, all of the prepositional phrases in the dative forms have been treated as verb arguments, i.e. the verb subcategorises for a particular type of prepositional phrase. However, it is not clear that prepositional phrases should always be treated as verb arguments. Jackendoff (1990, ch. 9) has argued that some of this information should be handled by adjunct rules which allow integration of information contributed by these phrases without the use of subcategorisation. He bases this claim on the fact that often these phrases make parallel contributions to the semantics of verbs which clearly do not subcategorise for them. Consider, for example, the sentences in (3.7).

- (3.7) a. Bill removed the rubbish for Helen.
b. Bill bought a bike for Helen.

- (3.8) a. Bill removed the rubbish.
b. Bill bought a bike.

Each of these sentences specifies that Bill does something for Helen's benefit. The preposition *for* is what Jackendoff refers to as the “*for* of beneficiary”. Since each of the corresponding sentences in (3.8) is perfectly grammatical without the prepositional phrase and the contribution of the prepositional phrase can be determined independently of the rest of the verb semantics, there does not seem to be any reason to assume that it is optionally subcategorised for. The adjunct rule which Jackendoff introduces to handle the semantics of this can be stated in the current representation as shown in (3.9)⁴.

- (3.9) If V corresponds to $\left[\begin{array}{c} \text{AFF} (X, \dots) \\ \{Set_Thematic\} \end{array} \right]$
and NP corresponds to Y,
then $[_S \dots [_{VP} V \dots [_{PP} \text{for NP}] \dots] \dots]$ may correspond to
 $\left[\begin{array}{c} \text{AFF} (X, \dots) \\ \left\{ \begin{array}{l} Set_Thematic, \\ for_to (\text{AFF}^+ (X, Y)) \end{array} \right\} \end{array} \right]$

There is, however, a difference between these sentences. (3.7a) cannot dativise, while (3.7b) can, as shown in (3.10). Jackendoff also specifies the semantics of the double object form through an adjunct rule, as shown in (3.11), but the rule only handles the semantics of the verbs of creating and so clearly cannot account for the dativisable nature of *buy*.

- (3.10) a. *Bill removed Helen the rubbish.
b. Bill bought Helen a bike.

⁴Note that the structure presented in this adjunct rule is different from that assumed for “for”-datives in the current work, as shown in (2.16), specifically in the fact that the present implementation assumes that the “for”-datives indicate that the direct object is intended to be HAD by the beneficiary. Thus it may be possible that there are several different types of “for of beneficiary” rules necessary.

- (3.11) If V corresponds to $\left[\begin{array}{c} \text{CREATE/PREPARE (X, Y)} \\ \{Set_Thematic\} \end{array} \right]$
 and NP corresponds to Z,
 then $[_S \dots [_{VP} V NP \dots] \dots]$ may correspond to
 $\left[\begin{array}{c} \text{CREATE/PREPARE (X, Y)} \\ \left\{ \begin{array}{c} Set_Thematic \\ for_to (AFF^+ (Y, Y)) \end{array} \right\} \end{array} \right]$

The problem with this solution is analogous to the problem with narrow-range lexical rules based on verb types (to be discussed in Section 3.3.4). It requires identification of a verb's type, the narrow class to which it belongs, in addition to the formation of a semantic representation for the verb. It seems unlikely that the lexicon is organised around these types, but rather is more likely that these types are a side-effect of the existence of groups of verbs with similar semantic structure. The solution could be salvaged, however, by modifying the adjunct rules such as that in (3.11) to specify more precisely the semantics of the V, rather than just mentioning that the verb has to be one of creating/preparing. They would thus have the same shape as the lexical rules presented in this work, but would exist at a level outside of the lexicon, in *extralexical* rules. The incorporation of extralexical rules into HPSG would entail adding many principles which would have to be checked for applicability at each level of semantic integration. The analysis of the dative alternation as presented here could be generalised to a group of extralexical rules controlling the integration of semantic information. The advantages and disadvantages of such a change, particularly with regard to the issue of true verb argument vs. adjunct, is a problem for further investigation.

3.3.2 Selectional Restrictions and the Double Object Form

The discussion in Section 3.1.2 mentions “compatibility with a particular notion”. This is a difficult idea to formalise. There is a fundamental problem with the current treatment of the dative alternation which stems from the problems discussed in Section 1.2.9 of handling selectional restrictions. In the double object form, the direct object must be capable of HAVING something, and the indirect object must be capable of being HAD (although normally HAVE corresponds to possession, and will be treated as such in the following discussion, as mentioned in footnote 4 it could also entail a more abstract relation). Thus although the *syntactic* forms in all of the sentences in (3.12) are theoretically correct for the double object form (that is, $NP_1 V NP_2 NP_3$) the *semantic* interpretation provided for (3.12b) would be incorrect because kids are not generally thrown, and that for (3.12c) would be incorrect because both the ball is incapable of possessing something, and kids are not generally possessed by inanimate things.

- (3.12) a. She throws the kids the ball.
 b. *She throws John the kids.
 c. *She throws the ball the kids.

Of course, no semantic interpretation for these sentences should be possible, and hence they *should* be rejected by the system. The problem lies in determining cognitive compatibility with the notions of “a possessor” and “something capable of being possessed”. It is psychologically unrealistic to assume that everything capable of being a possessor has a property in its property list specifying “possessor”. Were this the case, the lexical entry for a noun would

need to specify every possible semantic role the noun could play. It is not enough to specify that any *animate* objects can be possessors, because non-animate entities such as groups or organisations (e.g. *a charity*, *the Boy Scouts*) can be possessors. Example (3.12c) is semantically anomalous because *the kids* are not typically objects which can be thrown. This is even more difficult to represent than “possessor”. This requires world knowledge about objects which can physically be handled (relevant to such verbs as *hand*, *pass*, *carry*, etc.). This type of information cannot be encapsulated in property lists or simple selectional restrictions. The solution to the problem is not, however, obvious and will not be addressed here.

3.3.3 Lexical Issues

Optional Arguments

Verbs often have optional arguments. These are elements central to the meaning of the verb which may either be explicitly defined in the use of the verb in a sentence, or left to be determined by context. For example, both *Sue drinks the martini* and *Sue drinks* are grammatical. In the latter, Sue’s drink is left unspecified, yet it is still implicit that she did drink something. What precisely it was will have to be filled in from context or previously specified information.

It is not possible to handle such optional arguments within the framework of HPSG. Due to the mechanism of the Subcategorisation principle, for which each element on the subcat list must be “checked off” in turn, the subcat list must be explicitly specified for each lexical entry. Therefore, each possible variation on the subcat list – i.e. with the optional argument and without – must appear in a separate lexical entry.

Scaling Up the Lexicon

It requires much effort to “scale up” the size of the lexicon in this implementation. Each semantic representation is dependent on a linguistic analysis of the “grammatically relevant” information it conveys. Furthermore, it is often difficult to characterise verbal semantics in terms of the function predicates available in the representation. Many attempts at such characterisation within the current framework seem somewhat forced and unnatural.

3.3.4 An Enriched Type Hierarchy vs. Enriched Semantics

HPSG is required to be explicit as to what ontological categories of linguistic objects are assumed and mathematically rigorous as to what structures are used to model them. “The role of the linguistic theory is to give a precise specification of which feature structures are to be considered admissible; the types of linguistic entities that correspond to the admissible feature structures constitute the predictions of the theory” (Pollard and Sag 1994, p. 8). The theory relies on a fully sorted type hierarchy which determines the attribute labels that any feature structure can have. “There is one sort symbol for each basic type (ontological category) of construct” (ibid., p. 17).

One of the primary uses of this type signature is to factor out word-related information which can be predicted from its type, reducing the amount of information which then needs to be stipulated in individual lexical entries (Pollard and Sag, 1987). It would therefore be valid within the HPSG perspective to include narrow verb classes as subtypes of the more general verb forms. This could be done, for example, by enriching the *vform* subtype

bse to have subtypes corresponding to the different verb classes, such as **communication**, **future_having**, etc. This change in the type signature would simplify the task at hand immensely. The dative alternation could then simply be handled by two lexical rules in the shape of Pinker’s broad-range lexical rules, one for “to”-datives and one for “for”-datives. The only additional criteria that needs to be specified would be the type of the verb’s **vform** head feature, which would consist of a disjunction of the dativisable **vform** subtypes. Complex semantic representation criteria would become superfluous, as verbs need only have a feature specifying it as an alternating verb in order to alternate. Clearly, however, it is unrealistic to assume that the cognitive representation of verbs contains such a feature or that verbs are organised around artificial types corresponding to semantic and/or syntactic subclasses. Even if it could be shown that verbs **are** organised around such types, the problem of explaining the cognitive process of deciding into what verb class a verb fits when acquiring verb semantics would remain.

The type signature could also be used to formally associate the core semantic structures of “to”- and “for”-datives, introduced in (2.15) on page 43, with verbs. The **loc** type could be further broken down to include types for specific verb constructions, such as **{to_dative, to_dbl_obj, for_dative, etc.}**, for which the subcategorisation list and the core elements of the semantics would be specified by their type. Only variations on the core elements would then need to be specified in the lexicon. This enrichment of the type hierarchy again seems inappropriate given the current attempt to remain psychologically relevant. The core semantic configuration among verbs in particular syntactic constructions is identified through analysis of the semantics of those verbs; it is unlikely that it exists external to the lexicon, or that there is some mechanism for enforcing that all lexical entries of verbs with a particular argument structure must contain the core configuration. The semantic core is identified from external analysis of the lexicon, not imposed upon the lexicon by an internal mechanism.

3.3.5 Unification and Defaults

In a unification based system such as the Attribute Logic Engine (ALE), it is difficult to judge whether one feature structure subsumes another – either the two feature structures unify or they don’t, but there is no mechanism for determining whether one feature structure is more general than the other. There are, however, cases where subsumption is relevant to some operation on lexical entries.

For example, the predicates which take a manner argument must explicitly specify a particular manner or “no_manner”. It does not, however, seem highly psychologically realistic to assume that simply because there are some verbs that specify a manner for one of its occurrences or sub-occurrences, there will be a similar manner field for verbs that don’t specify any. It seems more reasonable to assume that a verb need not specify a manner or mention a manner field at all if such a manner is not relevant to its semantics. Any lexical rules or other operations which depend on the existence of that manner field should simply treat the manner field of a function which has no manner specified as “no_manner” by default. This entails treating a lexical entry for which the manner field is unspecified as more general than lexical entries for which the manner field is specified, and requiring operations to be performed only on the more specific entries.

Unfortunately, such a treatment is not possible due to a basic duality between the declarativeness required by the type hierarchy and specification of defaults. For our example, the relevant bit of the type hierarchy is shown in (3.13).

```

manner sub [no_manner, yes_manner].
  no_manner sub [].
  yes_manner sub [carrying, eating, grabbing, handing, lifting,
                  pulling, shouting, throwing, whispering, winning].
(3.13)      carrying sub [].           eating sub [].
            grabbing sub [].          handing sub [].
            lifting sub [].           pulling sub [].
            shouting sub [].          throwing sub [].
            whispering sub [].        winning sub [].

```

A lexical rule which requires that the specification of the feature **manner** for a particular function be **manner** will unify with any manner, whether it be **no_manner**, **yes_manner**, or any subtype thereof. There is no way to indicate that if a particular lexical entry is not explicit about what subtype of manner the function expresses (i.e. the value of the manner feature will be **manner**), then the lexical rule should treat the entry as if it specified **no_manner** explicitly due to the unification process. **Manner** is a more general type than **yes_manner**, and so they unify. There is no indication that **manner** also subsumes **yes_manner**, which would be needed to treat the value **yes_manner** differently from the generic value **manner** in a lexical entry. There is no way around the required explicit specification of **no_manner** for all function predicates in the representation of verb semantic which don't involve a manner. Realistically these predicates should be treated as having default values which can be overridden in the exceptional cases in which specific manners need to be indicated.

3.4 Semantic Issues

3.4.1 Compositionality

The discussion in Section 3.3.1 raises the issue of compositionality. The current approach is clearly not compositional, since the semantic interpretation is dictated by the verb semantics.

3.4.2 Inferencing

The effectiveness of a semantic system can really only be evaluated in terms of the inferences which it allows. Work in the Jackendoff/Pinker vein, such as the current implementation, relies on inference patterns associated with the various functions, as introduced on page 17, in the form of schemas or rules describing valid inferences. The inference rule (1.25) is repeated here as (3.14).

```

(3.14)  At the termination of [GO (X, [to (Y)])],
        it is the case that [BE (X, [at (Y)])].

```

According to Jackendoff (1990, p. 39), “Each element in a lexical decomposition can be regarded as that item’s access to more general-purpose rules of inference.” Inference rules are therefore defined over the substructures of semantic structure.

Jackendoff (1990) proposes that certain inference rules apply generally to particular conceptual structures, while other ones are dependent on specified semantic field. To quote his example of a field-specific inference pattern and the ensuing discussion of inference patterns,

in the spatial field, one fundamental principle stipulates that an object cannot be in two disjoint places at once. From this principle plus rule [(3.14)], it follows that an object that travels from one place to another is not still in its original position. However, in the field of information transfer, this inference does not hold. If Bill transfers information to Harry, by [(3.14)] we can infer that Harry ends up having the information. But since information, unlike objects, can be in more than one place at a time, Bill still may have the information too. Hence rule [(3.14)] generalises from the spatial field to information transfer, but the principle of exclusive location does not. Thus inference rules as well as lexical entries benefit from a featural decomposition of concepts: the Thematic Relations Hypothesis⁵ and the use of the semantic field feature permit us to generalise just those aspects that are general, while retaining necessary distinctions. (Jackendoff 1990, p. 27)

The mechanisms of inferencing have not been included in the current implementation. I will remain neutral on how these inferences occur. If it is via the schemata-type rules which Jackendoff proposes, it should be possible to define the semantic consequence relations directly over feature structures. The details of this as well as any implications for/interactions with the HPSG formalism will be left to further investigation.

3.4.3 Quantification

Zwarts and Verkuyl (1994, p. 8) discuss Jackendoff’s weaknesses in handling referential and quantificational properties. They point out that all the sentences in (3.15a)–(3.15d) would be represented as (3.15e) since all noun phrases containing the noun *man* would be represented as $[_{THING} man]$.

- (3.15) a. Every man walked to Rome.
 b. A man walked to Rome.
 c. The man walked to Rome.
 d. At least one man walked to Rome.
 e. $[_{EVENT} GO ([_{THING} man], [_{PATH} to ([_{PLACE} Rome)])]$

Although Zwarts and Verkuyl do not propose a definitive solution to these problems, they do show that “a proper formalisation of Jackendoff’s [Conceptual Semantics (CS)] might very well pave the way for the incorporation of quantification in CS.’, and provide a few options for treating it within their formalisation.

The current implementation adopts Jackendoff’s treatment of noun phrases, suggesting that it would also suffer from the same weaknesses. However, HPSG provides a mechanism for handling quantifiers through the Semantics Principle and the Quantifier Binding Condition, as described in Pollard and Sag 1994, chapter 8. Although the quantification is not explicitly represented within the semantic representation of feature structure representing a sentence, it is marked within the `CONTEXT` field, in a subfield called `QUANTS`. The effect of quantification on semantic inferencing and the adequacy of the current treatment of quantification for handling such inferencing have not been explored in this work and will not be discussed.

⁵See page 4 for introduction.

Chapter 4

Conclusions

Representation of the semantics of individual lexical items is a critical component of any language processing system hoping to achieve human-level competence in language understanding. It is, however, an extremely difficult task given the interaction of lexical knowledge with more general conceptual knowledge, including representations from the perceptual domains.

A good place to start, then, is to work on representing just those elements of semantics which have relevance to syntax. These grammatically relevant elements are more accessible for observation than their grammatically irrelevant counterparts, as they can be identified through linguistic analysis showing syntactic effects of variations in particular elements of meaning. Furthermore, these grammatically relevant semantic criteria can be used to constrain the size of the lexicon. This is done via lexical rules, which define syntactic alternations in terms of their semantic implications.

Such rules have been defined and shown to effectively capture the dative alternation in this thesis. The existence of the rules in the lexicon means that only one lexical entry needs to be defined for most verbs. The lexical rules capture generalisations about the semantic meaning associated with particular syntactic forms. When applied to verbs in the lexicon, they define the semantic shift associated with the syntactic shift the verb argument structure undergoes. These rules utilise precise semantic criteria to determine which verbs are able to undergo the defined semantic and syntactic shift.

The grammatically relevant semantic criteria have been formed into a formal representation, largely following that used by Stephen Pinker. Pinker, however, was heavily influenced by Ray Jackendoff, adopting many elements of his analysis without modification, and thus both Pinker's and Jackendoff's approaches have been discussed in some detail. Differences in the two theories have been identified, and the implemented representation combines insights from each. It includes a principled lexical decomposition and recursive mechanisms for structuring semantic representations.

4.1 Jackendoff vs. Pinker

Jackendoff's ultimate goal is to represent all conceptual knowledge through identification of the primitives and combinatorial principles which compose "mental information structures". He views language as "a relatively overt realization of conceptual structure" (Jackendoff 1990, p. 18), and thus relies upon linguistic analysis for determination of the conceptual primitives.

Pinker, on the other hand, has a much more focused goal: the modelling of the semantics of predicates and arguments relevant to syntactic argument structure and the role of the semantics in an explanation of child acquisition of argument structure. His analysis is therefore constrained to identifying “grammatically-relevant” semantic elements, rather than attempting to discover the set of elements with which *all* concepts are represented. He emphasises that the semantic criteria on which his lexical rules are based are not dependent on cognitively salient features of verb semantics, pertaining to characteristic or typical features of the events in the world that a verb can refer to, but on which aspects of the events are constrained by the verb’s semantic structure.

The differences in focus of the two theories discussed in this thesis manifest themselves in the depth to which each addresses systematic semantic relations across syntactic structures. Jackendoff proposes a few rules which reflect the semantic contributions of certain elements of syntactic constructions, capturing the semantic intuitions associated with these syntactic forms for some cases. There are clear limitations, however, to the applicability of these rules and doubts about their cognitive relevance as structured in (Jackendoff 1990). Furthermore, Jackendoff provides no mechanism for relating different argument structures of a single verb, or for formally indicating why certain verbs cannot appear in particular syntactic forms. Thus Jackendoff can generate an appropriate semantic representation of a sentence in many cases, given a lexical entry specifying the syntactic form in which it appears, but he cannot provide a method for determining which verbs can appear in such a syntactic form.

In contrast, Pinker’s entire framework is geared at providing just such a method. He identifies semantic criteria differentiating verb subclasses as to their syntactic behaviour and defines explicit rules governing the alternation of a verb from one syntactic form to another. These rules apply only to verbs which meet certain semantic criteria, and indicate precisely the semantic consequences of the syntactic alternation.

It is clear that for this thesis’ goal of capturing verbal syntactic alternations, Pinker’s observations are more directly relevant than Jackendoff’s. Pinker’s analysis forms the basis for the lexical rule definitions as implemented in the current work. However, Jackendoff’s work remains extremely important, as his justification for the primitive elements in the form of linguistic analysis is often more thorough than that of Pinker.

4.2 Reduction of Lexical Redundancy

Probably the most important result of this work in computational terms is that of reduction in the size of the lexicon. By capturing generalisations about syntactic forms in terms of semantic criteria and semantic consequences, only one lexical entry needs to be specified for each verb which alternates in standard ways. As long as the basic form of the verb is defined appropriately, the system itself can determine whether the verb should appear in other syntactic and semantic forms via its lexical rules.

4.3 Psychological Validity

The basis of the representation in Jackendoff’s linguistic analysis, combined with Pinker’s justifications in terms of psychological studies, is central to the claim that this representation has cognitive relevance. The set of primitive functions used in the representation definition

is not an ad hoc solution to the problem of semantic representation, but rather is a solution motivated by psycholinguistic investigation.

In addition, the analysis of syntactic alternations in terms of lexical rules is psychologically productive. It enables generalisation of argument structure to newly learned verbs, given a semantic structure for the verb. That is, when a particular semantic structure has been acquired for a verb, there is a clear mechanism for determining whether the verb will participate in syntactic alternations. Pinker has shown that this happens when children learn new verbs – the children will only use verbs in both of two alternating syntactic constructions if they understand the semantics of the verb to be compatible with the semantics of classes of verbs which alternate. Thus lexical rules model this compatibility determination. It is a much more cognitively realistic approach than the organisation of verbs around artificial types which are determined according to semantic similarity without any formal definition of what constitutes a member of the type.

4.4 Topics for Further Research

- **Temporal Representation**

The temporal representation as utilised in this thesis is adequate for the task of modelling the dative alternation. It clearly does not, however, capture the full range of temporal distinctions relevant to verb semantics. Further work on this is necessary.

- **Nominal Representation**

Until a comprehensive semantics for all major categories of words has been developed, it will be impossible to fully define the composition of words in a sentence. It is important to take into account how the semantics of a nominal contributes to a verb's meaning. Consider as an example the sentences *John baked the potato* and *John baked the cake*. In the current approach, the two senses of *bake* — that of changing the state of the potato and that of creating the cake — would require separate lexical entries. Pustejovsky (1991) points out, however, that if the semantics of the noun are taken into account and integrated with an appropriate representation of the verb, the different senses could be a result of the argument/verb interaction. It would be interesting to see whether Pustejovsky's approach could be incorporated into the current framework.

- **Property Representation**

The question of how to enforce selectional restrictions and the associated issue of cognitive compatibility of particular properties with the representation of a nominal have come up several times in this thesis. It is a topic deserving of further investigation, but a difficult one to address. It depends on the representation of nominals, and on integration of situation-dependent information and world knowledge with semantic representation.

- **Linking theory**

The mapping from semantic structure to syntactic form is a crucial element of both Pinker's and Jackendoff's theories. However, in this implementation their use of linking rules has been eschewed in favour of the mechanism of HPSG. There does seem to be a consistent relationship between semantic argument positions and syntactic argument positions, however. It would be interesting to see whether some type of linking theory could be developed within HPSG, and what effects this would have on the HPSG principles.

Appendix A

Glossary

argument structure	the information that specifies how a verb's arguments are encoded in syntax.
causative alternation	a syntactic phenomenon in which intransitive verbs can be transformed into causatives, i.e. <i>open</i> which can alternate between the form <i>The book opened</i> and <i>I opened the book</i> .
conceptual structure	a representation of the semantics of a concept.
conflation class	a set of possible predicates in a language defined by the thematic core of an argument structure.
dative alternation	a syntactic phenomenon in which verbs such as <i>give</i> can alternate between the form <i>John gave the book to Bill</i> and <i>John gave Bill the book</i> .
dativisable	a verb is dativisable if it can occur in both forms of the dative alternation.
lexical entry	representation of morphological, phonological, syntactic (part of speech and argument structure) and semantic information pertaining to a word.
lexical rule	a rule which specifies a mapping between one semantic structure and another.
linking rule	a rule which specifies how a semantic structure is mapped to syntactic argument structure.
locative alternation	a syntactic phenomenon in which verbs such as <i>splash</i> can alternate between the form <i>John splashed water on the dog</i> and <i>John splashed the dog with water</i> .
nondativisable	a verb is nondativisable if it only occurs in one form of the dative alternation.
passive alternation	a syntactic phenomenon in which verbs such as <i>touch</i> can alternate between the form <i>John touched the wall</i> and <i>The wall was touched by John</i> .

Appendix B

Representational Inventories

B.1 Jackendoff

Conceptual Constituents

EVENT PATH
STATE THING
PLACE

Functions expanding Conceptual Constituents

GO STAY
BE ORIENT
EXT INCH
CONF MOVE
AFF REACT
EXCH CAUSE (CS)
place-functions (*at, on, in, under, ...*)
path-functions (*to, from, toward, away-from, via*)

Subordinating Relations

FROM BY
WITH FOR
EXCH

Other mechanisms/notations

- Action and Thematic tiers
- Adjunct Rules
- features on functions
- \pm, u features: function occurs positively, negatively, or undetermined/neutrally
- semantic field features *function subscripts indicating the semantic field in which to interpret conceptual constituents: temporal, possession, identificational, circumstantial, existential*
- coindexing *subscripts used within lexical entries indicating a correspondence between syntactic and semantic/conceptual argument positions*
- optionality marking *various notations are used to indicate optional elements in conceptual structures. See Jackendoff 1990, ch. 4.*

B.2 Pinker

Conceptual Constituents

EVENT PATH
STATE PROPERTY
THING MANNER
PLACE

Functions expanding Conceptual Constituents

{ \pm dynamic, \pm control}

yielding predicates labelled

GO ACT BE HAVE
place-functions (*at, on, in, under, ...*)
path-functions (*to, into, toward, ...*)

Features of subordinating relations

- cause-focus versus effect-focus
- success versus failure
- occurrence versus nonoccurrence
- purposive
- deontic

Properties

$$\left(\begin{array}{c} animate \\ inanimate \end{array} \left(\begin{array}{c} \left(\begin{array}{c} human \\ nonhuman \end{array} \right) \\ \left(\begin{array}{c} 0D \\ 1D \\ 2D \\ 3D \end{array} \right) \end{array} \right) \left(\begin{array}{c} count \\ mass \end{array} \left(\begin{array}{c} \left(\begin{array}{c} rigid \\ flexible \end{array} \right) \\ \left(\begin{array}{c} substance \\ aggregate \rightarrow \end{array} \right) \end{array} \right) \left(\begin{array}{c} \left(\begin{array}{c} liquid \\ semisolid \end{array} \right) \\ Parts \\ \downarrow \\ PROPERTY \end{array} \right) \end{array} \right) \right)$$

Temporal objects

time-line point region

Other mechanisms/notations

- Lexical Rules
- semantic field annotation *Annotation on STATES and EVENTS indicating the semantic field in which to interpret conceptual constituents: possessional, physical, perceptual, epistemic, social, intrapsychic, responsibility, psychological*
- quoted constants *used for unelaborated conceptual information*
- coindexing *variables X, Y, Z used to express coreference between conceptual constituents at different argument positions within the semantic structure*
- open arguments *conceptual constituents which can be linked to a syntactic role in a verb's argument structure¹*

¹Open arguments are notated with brackets [] in Pinker's semantic representations

B.3 Verspoor

Conceptual Constituents

EVENT PATH
STATE PROPERTY
THING MANNER
PLACE

Functions expanding Conceptual Constituents

{±event, ±dynamic, ±control}

yielding predicates labelled

GO STAY MOVE ORIENT
BE HAVE AFF

place-functions (*at, on, in, under, ...*)

path-functions (*to, into, toward, ...*)

Features of subordinating relations

- cause-focus versus effect-focus
- success versus failure
- occurrence versus nonoccurrence
- purposive
- deontic

Properties

$$\bullet \left(\begin{array}{c} \text{animate} \left(\begin{array}{c} \text{human} \\ \text{nonhuman} \end{array} \right) \\ \text{inanimate} \left(\begin{array}{c} 0D \\ 1D \\ 2D \\ 3D \end{array} \right) \\ \left(\begin{array}{c} \text{count} \left(\begin{array}{c} \text{rigid} \\ \text{flexible} \end{array} \right) \\ \text{mass} \left(\begin{array}{c} \text{substance} \left(\begin{array}{c} \text{liquid} \\ \text{semisolid} \\ \text{Parts} \end{array} \right) \\ \text{aggregate} \rightarrow \text{PROPERTY} \end{array} \right) \end{array} \right) \end{array} \right)$$

- Semantic conditions on nominal objects

Other mechanisms/notations

- Action and Thematic tiers
- Lexical Rules
- semantic field features *function subscripts indicating the semantic field in which to interpret conceptual constituents: epistemic, perceptual, physical, possessional, psychological, spatial, existential*
- quoted constants *used for unelaborated conceptual information*
- time points *used to indicate that an event occurs instantaneously; continuous events are notated with a value “continuous” in the representation of time*

Appendix C

Lexical Entries

C.1 Specified Lexical entries

Verbs of Giving

<div> <div>give</div> <div>V</div> <div>NP₁, NP₂, to NP₃</div> <div> <div>AFF (THING₁, THING₂, time₀, no_manner)</div> <div> <div> <div><i>effect</i></div> <div>(GO_{possession} (THING₂, to (at (THING₃))), time₀, no_manner))</div> </div> </div> </div> </div>
<div> <div>pass</div> <div>V</div> <div>NP₁, NP₂, to NP₃</div> <div> <div>AFF (THING₁, THING₂, time₀, no_manner)</div> <div> <div> <div><i>effect</i></div> <div>(GO_{phys_custody} (THING₂, to (at (THING₃))), time₀, no_manner))</div> </div> </div> </div> </div>
<div> <div>hand</div> <div>V</div> <div>NP₁, NP₂, to NP₃</div> <div> <div>AFF (THING₁, THING₂, time₀, handing_manner)</div> <div> <div> <div><i>effect</i></div> <div>(GO_{phys_custody} (THING₂, to (at (THING₃))), time₀, no_manner))</div> </div> </div> </div> </div>
<div> <div>{send, mail, ship}</div> <div>V</div> <div>NP₁, NP₂, to NP₃</div> <div> <div>AFF (THING₁, THING₂, time₀, no_manner)</div> <div> <div> <div><i>effect</i></div> <div>(GO_{possession} (THING₂, to (at (THING₃))), time₂, no_manner)),</div> <div><i>means</i></div> <div>(AFF (THING^{postal service}, THING₂, time₁, no_manner))</div> </div> </div> </div> </div>

sell V NP ₁ , NP ₂ , to NP ₃ $\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner})), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_3, \text{THING}_4^{\text{money}}, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_4, \text{to}(\text{at}(\text{THING}_1)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right]$
trade V NP ₁ , NP ₂ , to NP ₃ $\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner})), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_3, \text{THING}_4, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_4, \text{to}(\text{at}(\text{THING}_1)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right]$
pay V NP ₁ , NP ₂ , to NP ₃ $\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2^{\text{money}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$
serve V NP ₁ , NP ₂ , to NP ₃ $\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2^{\text{consumable}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{AFF}(\text{THING}_3, \text{THING}_2, \text{time}_1, \text{eating_manner})) \end{array} \right\} \end{array} \right]$
feed V NP ₁ , NP ₂ , to NP ₃ $\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2^{\text{consumable}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{AFF}(\text{THING}_3, \text{THING}_2, \text{time}_1, \text{eating_manner})) \end{array} \right\} \end{array} \right]$

Verbs of Communication

tell	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2 \textit{for_to}(\text{BE}_{\text{epistemic}}(\text{THING}_2, \textit{at}(\text{THING}_3))), \text{time}_0) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{communication}}(\text{THING}_2, \textit{to}(\textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
ask	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2 \textit{for_to}(\text{BE}_{\text{epistemic}}(\text{THING}_2, \textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{communication}}(\text{THING}_2, \textit{to}(\textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner}), \\ \textit{for_to} \\ (\text{GO}_{\text{communication}}(\text{THING}_4^{\textit{answer}}, \textit{to}(\textit{at}(\text{THING}_1))), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
radio	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{communication}}(\text{THING}_2, \textit{to}(\textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner}), \\ \textit{means} \\ (\text{AFF}(\text{THING}_1, \text{THING}^{\textit{radio}}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	

Verbs of Manner of Speaking

shout	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2^{\textit{sound}}, \text{time}_0, \text{shouting_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{perceptual}}(\text{THING}_2, \textit{to}(\textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
whisper	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{whispering_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{perceptual}}(\text{THING}_2, \textit{to}(\textit{at}(\text{THING}_3))), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	

Verbs of Future Having

bequeath	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_2, \text{no_manner})), \\ \text{means} \\ (\text{BE}(\text{THING}^{\text{will}}, \text{THING}^{\text{existence}}, \text{time}_0)) \end{array} \right\} \end{array} \right]$	
refer	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_1, \text{no_manner})), \\ \text{means} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{communication}}(\text{THING}_4^{\text{reference}}, \text{to}(\text{at}(\text{THING}_2)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right]$	

Verbs of Causation of Motion

throw	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{throwing_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
pull	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{continuous}, \text{pulling_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{continuous}, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
lift	
V	
NP ₁ , NP ₂ , to NP ₃	
$\left[\begin{array}{l} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{continuous}, \text{lifting_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{continuous}, \text{no_manner})) \end{array} \right\} \end{array} \right]$	

$ \begin{bmatrix} \text{carry} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{c} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{continuous}, \text{carrying_manner}) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \left[\begin{array}{c} \text{with}(\text{at}(\text{THING}_1)) \\ \text{to}(\text{at}(\text{THING}_3)) \end{array} \right], \text{continuous}, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{bmatrix} $
$ \begin{bmatrix} \text{bring} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{c} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{continuous}, \text{no_manner}) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \left[\begin{array}{c} \text{with}(\text{at}(\text{THING}_1)) \\ \text{to}(\text{at}(\text{THING}_3^{\text{here}})) \end{array} \right], \text{continuous}, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{bmatrix} $
$ \begin{bmatrix} \text{take} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{c} \text{AFF}_{\text{physical}}(\text{THING}_1, \text{THING}_2, \text{continuous}, \text{no_manner}) \\ \left\{ \begin{array}{c} \text{effect} \\ (\text{GO}_{\text{spatial}}(\text{THING}_2, \left[\begin{array}{c} \text{with}(\text{at}(\text{THING}_1)) \\ \text{to}(\text{at}(\text{THING}_3^{\text{there}})) \end{array} \right], \text{continuous}, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{bmatrix} $

Verbs of Fulfilling/Deserving

$ \begin{bmatrix} \text{present} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{with NP}_2 \\ \left[\begin{array}{c} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{c} \text{effect} \\ \left(\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner}) \right. \\ \left. \left. \text{for_to} \left(\begin{array}{c} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ (\text{EVENT}(\text{THING}_3)) \end{array} \right) \right) \right\} \end{array} \right] \end{bmatrix} $
$ \begin{bmatrix} \text{reward} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{with NP}_2 \\ \left[\begin{array}{c} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{c} \text{effect} \\ \left(\text{HAVE}(\text{THING}_3, \left\{ \begin{array}{c} \text{for_to} \left(\begin{array}{c} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ \text{EVENT}(\text{THING}_3) \end{array} \right), \text{reward} \end{array} \right\}, \text{time}_0) \right) \right\} \end{array} \right] \end{bmatrix} $

$\left[\begin{array}{l} \text{reward} \\ \text{V} \\ \text{NP}_1, \text{NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2^{\text{reward}}, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array} \right]$	
$\left[\begin{array}{l} \text{honor} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{ with NP}_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ \left(\text{HAVE}(\text{THING}_3, \right. \\ \left. \left. \left\{ \begin{array}{l} \text{for_to} \left(\begin{array}{l} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ \text{EVENT}(\text{THING}_3) \end{array} \right), \text{honor} \end{array} \right\}, \text{time}_0) \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]$	
$\left[\begin{array}{l} \text{honor} \\ \text{V} \\ \text{NP}_1, \text{NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2^{\text{honor}}, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array} \right]$	

Verbs of Obtaining

$\left[\begin{array}{l} \text{get} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{ for NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \end{array} \right] \end{array} \right]$	
$\left[\begin{array}{l} \text{buy} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{ for NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_4^{\text{money}}, \text{time}_2, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_4, \text{to}(\text{at}(\text{THING})), \text{time}_2, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]$	

grab	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF}_{\text{physical}} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{grabbing_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \end{array} \right]$	
win	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ (\text{AFF} (\text{THING}_1, \text{THING}^{\text{contest}}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
earn	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ (\text{AFF} (\text{THING}_1, \text{THING}^{\text{task}}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
order	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ \left(\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_4^{\text{orderee}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ \text{GO}_{\text{communication}} (\text{THING}^{\text{order}}, \text{to} (\text{at} (\text{THING}_4)), \text{time}_0, \text{no_manner}) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right]$	

find	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2^{\text{found}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \end{array} \right]$	
steal	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2^{\text{stolen}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \end{array} \right]$	

Verbs of Creating

make	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to} (\text{at} (\text{existence})), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \end{array} \right]$	
bake	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to} (\text{at} (\text{existence})), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ \text{AFF} (\text{THING}_1, \text{THING}^{\text{oven}}, \text{time}_0, \text{no_manner}) \end{array} \right\} \end{array} \right]$	
cook	
V	
NP ₁ , NP ₂ , for NP ₃	
$\left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to} (\text{at} (\text{existence})), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ \text{AFF} (\text{THING}_1, \text{THING}^{\text{cooker}}, \text{time}_0, \text{no_manner}) \end{array} \right\} \end{array} \right]$	

$ \left[\begin{array}{l} \text{sew} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to} (\text{at} (\text{existence})), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ \text{AFF} (\text{THING}_1, \text{THING}^{\text{sewing machine}}, \text{time}_0, \text{no_manner}) \end{array} \right\} \end{array} \right] \end{array} \right] $
$ \left[\begin{array}{l} \text{build} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2^{\text{aggregate: parts}_5}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to} (\text{at} (\text{existence})), \text{time}_0, \text{no_manner})), \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ \text{AFF} (\text{THING}_1, \text{THING}^{\text{parts}_5}, \text{time}_0, \text{no_manner}) \end{array} \right\} \end{array} \right] \end{array} \right] $

Verbs of Choosing

$ \left[\begin{array}{l} \{\text{choose, select, pick}\} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_2^{\text{preferred}}, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{cause} \\ \left(\text{AFF}^+ (\text{THING}_2, \text{THING}_1, \text{time}_0, \text{no_manner}) \right) \\ \left\{ \begin{array}{l} \text{cause} \\ (\text{AFF}^+ (\text{THING}_2, \text{THING}_3, \text{time}_2, \text{no_manner})) \end{array} \right\} \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_2)) \end{array} \right\} \end{array} \right] \end{array} \right] $
$ \left[\begin{array}{l} \{\text{favor, prefer}\} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{for NP}_3 \\ \left[\begin{array}{l} \text{AFF}^+ (\text{THING}_2^{\text{preferred}}, \text{THING}_1, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{cause} \\ (\text{AFF}^+ (\text{THING}_2, \text{THING}_3, \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right] $

C.2 Generated Lexical entries

Verbs of Giving

$\left[\begin{array}{l} \{\text{give, pass, hand}\} \\ V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array} \right]$
$\left[\begin{array}{l} \{\text{send, mail, ship}\} \\ V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_2)), \\ \text{means} \\ (\text{AFF}(\text{THING}^{\text{postal service}}, \text{THING}_2, \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$
$\left[\begin{array}{l} \text{sell} \\ V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_0), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_3, \text{THING}_4^{\text{money}}, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_4, \text{to}(\text{at}(\text{THING}_1)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]$
$\left[\begin{array}{l} \text{trade} \\ V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_0)), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_3, \text{THING}_4, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_4, \text{to}(\text{at}(\text{THING}_1)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]$
$\left[\begin{array}{l} \text{pay} \\ V \\ NP_1, NP_3, NP_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2^{\text{money}}, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array} \right]$

$$\left[\begin{array}{l} \{\text{serve, feed}\} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2^{\text{consumable}}, \text{time}_0)), \\ \text{for_to} \\ (\text{AFF} (\text{THING}_3, \text{THING}_2, \text{time}_1, \text{eating_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Verbs of Communication

$$\left[\begin{array}{l} \text{tell} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2^{\text{for_to}(\text{BE}_{\text{epistemic}}(\text{THING}_2, \text{at}(\text{THING}_3))), \text{time}_0)), \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\left[\begin{array}{l} \text{ask} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2^{\text{for_to}(\text{BE}_{\text{epistemic}}(\text{THING}_2, \text{at}(\text{THING}_3))), \text{time}_0)), \\ \text{for_to} \\ (\text{GO}_{\text{communication}} (\text{THING}_4^{\text{answer}}, \text{to}(\text{at}(\text{THING}_1)), \text{time}_1, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\left[\begin{array}{l} \text{radio} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_0)), \\ \text{means} \\ (\text{AFF} (\text{THING}_1, \text{THING}^{\text{radio}}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Verbs of Manner of Speaking

No lexical entries generated for *shout* or *whisper*.

Verbs of Future Having

$$\left[\begin{array}{l} \text{bequeath} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_1, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_2)), \\ \text{means} \\ (\text{BE} (\text{THING}^{\text{will}}, \text{THING}^{\text{existence}}, \text{time}_0)) \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\left[\begin{array}{l} \text{refer} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)), \\ \text{means} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{communication}}(\text{THING}_4^{\text{reference}}, \text{to}(\text{at}(\text{THING}_2)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Verbs of Causation of Motion

No lexical entries generated for *pull*, *lift*, or *carry*.

$$\left[\begin{array}{l} \text{throw} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{throwing_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\left[\begin{array}{l} \text{bring} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3^{\text{here}}, \text{continuous}, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{continuous})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\left[\begin{array}{l} \text{take} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3^{\text{there}}, \text{continuous}, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{continuous})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Verbs of Fulfilling/Deserving

No lexical entries generated for *reward* or *honor*.

$$\left[\begin{array}{l} \text{present} \\ \text{V} \\ \text{NP}_1, \text{NP}_2, \text{to NP}_3 \\ \left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_2 \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_3)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\text{for_to} \left(\begin{array}{l} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ (\text{EVENT}(\text{THING}_3)) \end{array} \right), \text{time}_0, \text{no_manner}$$

$$\left[\begin{array}{l} \text{present} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ \text{for_to} \left(\begin{array}{l} \text{BE}_{\text{possession}}(\text{THING}_2, \text{at}(\text{THING}_3)) \\ \text{fulfills} \\ (\text{EVENT} (\text{THING}_3)) \end{array} \right) \end{array} \right\} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_0)) \end{array} \right] \end{array} \right]$$

Verbs of Obtaining

$$\left[\begin{array}{l} \text{get} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \text{means} \\ (\text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\left[\begin{array}{l} \text{buy} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \text{means} \\ (\text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner})), \\ \text{obligates} \\ \left(\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_4^{\text{money}}, \text{time}_2, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_4, \text{to} (\text{at} (\text{THING})), \text{time}_2, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right] \end{array} \right]$$

$$\left[\begin{array}{l} \text{grab} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{grabbing_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}} (\text{THING}_2, \text{to} (\text{at} (\text{THING}_1)), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \text{means} \\ (\text{AFF} (\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

win	
V	
NP ₁ , NP ₃ , NP ₂	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_1)), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \text{means} \\ (\text{AFF}(\text{THING}_1, \text{THING}^{\text{contest}}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
earn	
V	
NP ₁ , NP ₃ , NP ₂	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_1)), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \text{means} \\ (\text{AFF}(\text{THING}_1, \text{THING}^{\text{task}}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	
order	
V	
NP ₁ , NP ₃ , NP ₂	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2, \text{to}(\text{at}(\text{THING}_1)), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \text{means} \\ \left(\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_4^{\text{orderee}}, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{communication}}(\text{THING}^{\text{order}}, \text{to}(\text{at}(\text{THING}_4)), \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right) \end{array} \right\} \end{array} \right]$	
find	
V	
NP ₁ , NP ₃ , NP ₂	
$\left[\begin{array}{l} \text{AFF}(\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{possession}}(\text{THING}_2^{\text{found}}, \text{to}(\text{at}(\text{THING}_1)), \text{time}_0)) \\ \text{for_to} \\ (\text{HAVE}(\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \text{means} \\ (\text{AFF}(\text{THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right]$	

Verbs of Creating

make	
V	
NP ₁ , NP ₃ , NP ₂	
$\left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to (at (existence))), time}_0)) \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \\ \text{means} \\ (\text{AFF (THING}_1, \text{THING}_2, \text{time}_0, \text{no_manner})) \end{array} \right]$	
bake	
V	
NP ₁ , NP ₃ , NP ₂	
$\left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to (at (existence))), time}_0)) \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \\ \text{means} \\ (\text{AFF (THING}_1, \text{THING}^{\text{oven}}, \text{time}_0, \text{no_manner})) \end{array} \right]$	
cook	
V	
NP ₁ , NP ₃ , NP ₂	
$\left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to (at (existence))), time}_0)) \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \\ \text{means} \\ (\text{AFF (THING}_1, \text{THING}^{\text{cooker}}, \text{time}_0, \text{no_manner})) \end{array} \right]$	
sew	
V	
NP ₁ , NP ₃ , NP ₂	
$\left[\begin{array}{l} \text{AFF (THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \text{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2, \text{to (at (existence))), time}_0)) \\ \text{for_to} \\ (\text{HAVE (THING}_3, \text{THING}_2, \text{time}_1)) \end{array} \right\} \\ \text{means} \\ (\text{AFF (THING}_1, \text{THING}^{\text{sewing machine}}, \text{time}_0, \text{no_manner})) \end{array} \right]$	

$$\left[\begin{array}{l} \text{build} \\ \text{V} \\ \text{NP}_1, \text{NP}_3, \text{NP}_2 \\ \left[\begin{array}{l} \text{AFF} (\text{THING}_1, \text{THING}_3, \text{time}_0, \text{no_manner}) \\ \left\{ \begin{array}{l} \textit{effect} \\ (\text{GO}_{\text{existential}} (\text{THING}_2^{\textit{aggregate: parts}_5}, \text{to} (\text{at} (\text{existence})), \text{time}_0)) \\ \textit{for_to} \\ (\text{HAVE} (\text{THING}_3, \text{THING}_2, \text{time}_1)) \\ \textit{means} \\ (\text{AFF} (\text{THING}_1, \text{THING}^{\textit{parts}_5}, \text{time}_0, \text{no_manner})) \end{array} \right\} \end{array} \right] \end{array} \right]$$

Verbs of Choosing

No lexical entries generated for *choose*, *pick*, *select*, *favor*, or *prefer*.

Appendix D

ALE Files

D.1 Type Hierarchy

```
bot sub [bool, case, cat, c_inds, conx,
        gend, head, ind, list, loc, marking,
        name, non_loc, non_loc_1, num, mod_synsem,
        per, pform, qfpsoa, sem_det, sem_obj, sign, set, vform,
        funcs, concept_constituents, polarity, time, sem_field].

bool sub [minus, plus].
  minus sub [].
  plus sub [].
case sub [nom, acc].
  nom sub [].
  acc sub [].
cat sub []
  intro [subcat:list_synsem,
        head:head,
        marking:marking].
c_inds sub []
  intro [addressee:ref,
        speaker:ref,
        utt_loc:ref].
conx sub []
  intro [backgr:set_psoa,
        c_inds:c_inds].
gend sub [fem, masc, neut].
  fem sub [].
  masc sub [].
  neut sub [].
head sub [func, subst].
  func sub [det, mark]
    intro [spec:synsem].
  det sub [].
  mark sub [].
subst sub [adj, noun, prep, reltvzr, verb]
  intro [prd:bool,
        mod:mod_synsem].
  adj sub [].
  noun sub []
    intro [case:case].
  prep sub []
    intro [pform:pform].
  reltvzr sub [].
  verb sub []
    intro [aux:bool,
          inv:bool,
          vform:vform].
mod_synsem sub [synsem, none].
```

```

synsem sub []
  intro [loc:loc,
        non_loc:non_loc].
none sub [].
ind sub [it, there, ref]
  intro [gen:gend,
        num:num,
        per:per].
it sub [].
there sub [].
ref sub [].
list sub [e_list, ne_list, list_quant, list_synsem, list_sign,
         list_props].
e_list sub [].
ne_list sub [ne_list_quant,
            ne_list_synsem,
            ne_list_sign,
            ne_list_props]
  intro [hd:bot,
        tl:list].
list_quant sub [e_list, ne_list_quant].
ne_list_quant sub []
  intro [hd:quant,
        tl:list_quant].
list_synsem sub [e_list, ne_list_synsem].
ne_list_synsem sub []
  intro [hd:synsem,
        tl:list_synsem].
list_sign sub [e_list, ne_list_sign].
ne_list_sign sub []
  intro [hd:sign,
        tl:list_sign].
list_props sub [e_list, ne_list_props].
ne_list_props sub []
  intro [hd:nom_obj,
        tl:list_props].
loc sub []
  intro [cat:cat,
        cont:sem_obj,
        conx:conx].
marking sub [marked, unmarked].
marked sub [comp, conj].
comp sub [for, that].
for sub [].
that sub [].
conj sub [].
unmarked sub [].
name sub [kim,sandy].
kim sub [].
sandy sub [].
non_loc sub []
  intro [inherited:non_loc_1,
        to_bind:non_loc_1].
non_loc_1 sub []
  intro [que:set_npro,
        rel:set_ref,
        slash:set_loc].
num sub [plur, sing].
plur sub [].
sing sub [].
per sub [first, second, third].
first sub [].
second sub [].
third sub [].
pform sub [verbal, nominal].
nominal sub [].

```

```

verbal sub [adjunctive, syntactic].
  adjunctive sub [in, with_accomp, for_temporal].
    in sub [].      with_accomp sub [].
    for_temporal sub []. % play piano for two hours
  syntactic sub [to, with, for_benefactive, for_trading].
    to sub [].      for_benefactive sub [].
    with sub [].    for_trading sub []. % buy something for $100
qfpsoa sub [property, un_relation, bin_relation, tri_relation,
  control_qfpsoa, top_level_desc, manner].
manner sub [no_manner, yes_manner].
  no_manner sub [].
  yes_manner sub [carrying, eating, grabbing, handing, lifting, pulling,
    shouting, throwing, whispering, winning].
    carrying sub [].      eating sub [].
    grabbing sub [].      handing sub [].
    lifting sub [].       pulling sub [].
    shouting sub [].      throwing sub [].
    whispering sub [].    winning sub [].
property sub [basic_prop, complex_prop].
  basic_prop sub [concept, animacy, dimension, count_or_mass].
  concept sub [gender, nom_prop]
    intro [inst:ref].
  gender sub [human, neuter].
  human sub [female, male].
    female sub [].      male sub [].      neuter sub [].
  nom_prop sub [answer, book, consumable, contest, cooker, difficult,
    found, honor, money, order, orderee, oven, parts, postal_service,
    preferred, radio, red, reference, reward, sewing_machine, sound,
    stolen, task, will].
    answer sub [].      book sub [].
    consumable sub [].  contest sub [].
    cooker sub [].      difficult sub [].
    found sub [].       honor sub [].
    money sub [].       order sub [].
    orderee sub [].     oven sub [].
    parts sub [].       postal_service sub [].
    preferred sub [].   radio sub [].
    red sub [].         reference sub [].
    reward sub [].      sewing_machine sub [].
    stolen sub [].      task sub [].
    will sub [].
  animacy sub [animate, inanimate].
  animate sub [human, nonhuman].
  nonhuman sub [].
  inanimate sub []
    intro [dim: dimension, comp: count_or_mass].
  dimension sub [zeroD, oneD, twoD, threeD].
    zeroD sub [].      oneD sub [].
    twoD sub [].       threeD sub [].
  count_or_mass sub [count, mass].
  count sub [rigid, flexible].
    rigid sub [].      flexible sub [].
  mass sub [substance, aggregate].
  substance sub [liquid, semisolid].
    liquid sub [].      semisolid sub [].
  aggregate sub []
    intro [parts: property].
  complex_prop sub [such_that_prop, for_to_prop]
    intro [state: state_desc].
    such_that_prop sub [].      for_to_prop sub [].
  un_relation sub [walk, run].
  walk sub [] intro [walker:ref].      run sub [] intro [runner:ref].
  bin_relation sub [see, hit, naming, composed_of, possess].
  see sub [] intro [seer:ref, seen:ref].
  hit sub [] intro [hitter:ref, hittee:ref].
  naming sub [] intro [bearer:ref, name:name].

```

```

    composed_of sub [] intro [composite:ref, composition:set_ref].
    possess sub [] intro [possessor:ref, possessed:ref].
    tri_relation sub [].
    control_qfpsoa sub [trying, tending, believing, persuading, bothering]
        intro [soa_arg:psoa].
    trying sub [] intro [tryer:ref].
    persuading sub [] intro [persuader:ref, persuaded:ref].
    tending sub [].
    believing sub [] intro [believer:ref].
    bothering sub [] intro [bothered:ref].
    sem_det sub [forall,exists,the].
    forall sub [].    exists sub [].    the sub [].
    sem_obj sub [nom_obj, psoa, quant].
    nom_obj sub [npro, pron, thing]
        intro [index:ind,
            restr:set_psoa].
    npro sub [].
    pron sub [ana, ppro].
    ana sub [recp, refl].
    recp sub [].
    refl sub [].
    ppro sub [].
    thing sub [existence, deictic]
        intro [index:ref].
    existence sub [].
    deictic sub [here_deictic, there_deictic].
    here_deictic sub [].
    there_deictic sub [].

    quant sub []
        intro [det:sem_det,
            restind:npro].
    psoa sub []
        intro [quants:list_quant,nucleus:qfpsoa].

    sign sub [word,non_word]
        intro [synsem:synsem,
            qstore:set_quant,
            qretr:list_quant].
    word sub [].
    non_word sub [trace,phrase,sentence].
    trace sub [].
    phrase sub [].
    sentence sub [].
    set sub [e_set, ne_set, set_loc, set_npro, set_psoa, set_quant, set_ref,
        set_thematic].
    e_set sub [].
    ne_set sub [ne_set_loc, ne_set_npro, ne_set_psoa, ne_set_quant,
        ne_set_ref, ne_set_thematic]
        intro [elt:bot, elts:set].
    set_loc sub [e_set, ne_set_loc].
    ne_set_loc sub []
        intro [elt:loc, elts:set_loc].
    set_npro sub [e_set, ne_set_npro].
    ne_set_npro sub []
        intro [elt:npro, elts:set_npro].
    set_psoa sub [e_set, ne_set_psoa].
    ne_set_psoa sub []
        intro [elt:psoa, elts:set_psoa].
    set_quant sub [e_set, ne_set_quant].
    ne_set_quant sub []
        intro [elt:quant, elts:set_quant].
    set_ref sub [e_set, ne_set_ref].
    ne_set_ref sub []
        intro [elt:ref, elts:set_ref].
    set_thematic sub [e_set, ne_set_thematic].

```

```

    ne_set_thematic sub []
        intro [elt:thematic, elts:set_thematic].
vform sub [bse, fin, ger, inf, pas, prp, psp].
    bse sub [].
    fin sub [].
    ger sub [].
    inf sub [].
    pas sub [].
    prp sub [].
    psp sub [].

polarity sub [bool, neutral].
    neutral sub [].

time sub [instantaneous, continuous].
    instantaneous sub [time_0, time_1, time_2].
        time_0 sub [].
        time_1 sub [].
        time_2 sub [].
    continuous sub [].

sem_field sub [epistemic, possessional, perceptual, physical, psychological,
    spatial, existential].
    epistemic sub [].
    existential sub [].
    perceptual sub [].
    physical sub [].
    possessional sub [communication, possession].
        communication sub [].
        possession sub [phys_custody].
            phys_custody sub [].
    psychological sub [].
    spatial sub [].

funcs sub [place_func, path_func, event_func, state_func, subord_func].
    subord_func sub [effect, cause, despite, but, let, prevent, means,
        for_to, obligates, fulfills].
        effect sub [].          cause sub [].          despite sub [].
        but sub [].            let sub [].            prevent sub [].
        means sub [].          for_to sub [].          obligates sub [].
        fulfills sub [].
    place_func sub [to_place, in_place, on_place, at_place, under_place,
        around_place].
        to_place sub [].      in_place sub [].      on_place sub [].
        at_place sub [].      under_place sub [].      around_place sub [].
    path_func sub [to_path, from_path, toward_path, away_from_path, via_path,
        with_accomp_path].
        to_path sub [].      from_path sub [].      toward_path sub [].
        away_from_path sub []. via_path sub [].      with_accomp_path sub [].
    event_func sub [go_func, stay_func, move_func].
        go_func sub [].      stay_func sub [].      move_func sub [].
    state_func sub [have_func, orient_func, aff_func, be_func].
        be_func sub [].      orient_func sub [].      have_func sub [].
        aff_func sub []
            intro [polarity:polarity].

concept_constituents sub [thing, occurrences, orientation, sem_desc].
    orientation sub [path, place].
        place sub []
            intro [afunc: place_func, arg1: thing].
        path sub [single_path, multi_path]
            intro [afunc: path_func, arg1: place].
            single_path sub [].
            multi_path sub [from_to_path, with_to_path]
                intro [afunc_2: path_func, arg_2: place].
                from_to_path sub []

```

```

    intro [afunc:from_path, afunc_2:to_path].
with_to_path sub []
    intro [afunc:with_accomp_path, afunc_2:to_path].

occurrences sub [one_ary_occ, two_ary_occ, event, state, manner_occ]
    intro [time:time, sem_field:sem_field].
manner_occ sub [one_ary_manner_occ, two_ary_manner_occ, manner_event]
    intro [manner:manner].
    one_ary_manner_occ sub [one_ary_manner_event].
    two_ary_manner_occ sub [two_ary_manner_event, manner_state].
one_ary_occ sub [one_ary_event, one_ary_manner_occ].
two_ary_occ sub [two_ary_event, state, two_ary_manner_occ]
    intro [arg2: concept_constituents].
state sub [be_state, have_state, orient_state, aff_state, manner_state]
    intro [afunc: state_func].
    be_state sub []
        intro [afunc: be_func, arg1:thing, arg2:place].
    have_state sub []
        intro [afunc: have_func, arg1:thing, arg2:thing, sem_field: possession].
    orient_state sub []
        intro [afunc: orient_func, arg1:thing, arg2:path].
    aff_state sub []
        intro [afunc: aff_func, arg1: thing, arg2: thing].
manner_state sub [aff_state].
event sub [two_ary_event, one_ary_event, manner_event]
    intro [afunc: event_func].
two_ary_event sub [go_event, stay_event, two_ary_manner_event].
    go_event sub []
        intro [afunc:go_func, arg1:thing, arg2:path].
    stay_event sub []
        intro [afunc:stay_func, arg1:thing, arg2:place].
    two_ary_manner_event sub [go_event].
one_ary_event sub [move_event, unspecified_event, one_ary_manner_event].
    move_event sub []
        intro [afunc: move_func, arg1: thing].
    unspecified_event sub []
        intro [afunc: event_func, arg1: thing].
    one_ary_manner_event sub [move_event].
manner_event sub [two_ary_manner_event, one_ary_manner_event].

sem_desc sub [description, thematic, func_category].
description sub [top_level_desc, other_desc].
top_level_desc sub [simple_sem, complex_sem]
    intro [prop:list_props, struct:other_desc].
simple_sem sub []
    intro [struct: occurrences].
complex_sem sub []
    intro [struct: event_desc].
other_desc sub [occurrences, complex].
complex sub [event_desc, state_desc]
    intro [action: occurrences, thematic: set_thematic].
event_desc sub []
    intro [action: aff_state].
state_desc sub []
    intro [action: state].

thematic sub []
    intro [afunc: subord_func, arg1: other_desc].

func_category sub [one_ary, occurrences]
    intro [afunc: funcs, arg1: concept_constituents].
one_ary sub [orientation, one_ary_occ, thematic].

```

D.2 Lexical Rules

```
% Lexical Rules
% =====

% Dative alternation Lexical Rules
% =====

% "to"-datives
% =====

% Basic "to"-dative lexical rule
%-----
basic_to_dative_lex_rule
(word,
  synsem:loc:((cat):(head:(verb, Verb,
    vform:bse),
    subcat:[ (@ np(Ind1)),
              (@ np(Ind2), @ case(acc)),
              (@ pp(to, Ind3))],
    marking:unmarked),
  cont:(nucleus:(complex_sem,
    struct:(action:(aff_state,
      afunc:aff_func,
      arg1: (Thing1, index:Ind1),
      arg2: (Thing2, index:Ind2, restr:Restr2),
      time: (Time0, instantaneous)),
      thematic: ThematicIn),
    prop:([Thing1, Thing2, Thing3])),
    quants:[]),
    conx:backgr:e_set),
  (@ empty_non_loc),
  qstore:e_set)
**>
(word,
  synsem:loc:((cat):(head:(verb, Verb),
    subcat:[ (@ np(Ind1)),
              (@ np(Ind3), @ case(acc)),
              (@ np(Ind2), @ case(acc))],
    marking:unmarked),
  cont:(nucleus:(complex_sem,
    struct:(action:(aff_state,
      afunc:aff_func,
      arg1: Thing1,
      arg2: (Thing3, index:Ind3),
      time: (Time0, instantaneous),
      manner: no_manner),
      thematic: ThematicOut),
    prop:([Thing1, Thing3, Thing2])),
    quants:[]),
    conx:backgr:e_set),
  (@ empty_non_loc),
  qstore:e_set)

if (simple_prop(Restr2),
  set_select((afunc:effect,
    arg1:(go_event,
```

```

        sem_field:(possession;spatial),
        afunc: go_func,
        arg1: Thing2,
        arg2:(single_path,
            afunc:to_path,
            arg1:(place,
                afunc:at_place,
                arg1: Thing3)),
        time: (Time1, instantaneous))), ThematicIn, Set),
union((elt:
    (afunc:effect,
    arg1:(have_state,
    afunc: have_func,
    arg1: Thing3,
    arg2: Thing2,
    time: (Time1, instantaneous))),
    elts:e_set), Set, ThematicOut))

morphs
X becomes X.

simple_prop(e_set) if true.
simple_prop((elt:(nucleus:basic_prop), elts:e_set)) if true.

% Communication Class lexical rule
%-----
communication_class_dative lex_rule
(word,
    synsem:loc:((cat):(head:(verb, Verb,
        vform:bse),
        subcat:[ (@ np(Ind1)),
            (@ np(Ind2), @ case(acc)),
            (@ pp(to, Ind3))],
        marking:unmarked),
    cont:(nucleus:(complex_sem,
        struct:(action:(aff_state,
            afunc:aff_func,
            arg1: (Thing1, index:Ind1),
            arg2: (Thing2, index:Ind2),
            time: (Time, instantaneous),
            manner: no_manner),
            thematic:ThematicIn),
        prop:([Thing1, Thing2, Thing3])),
        quants:[]),
        conx:backgr:e_set),
    (@ empty_non_loc),
    qstore:e_set)
**>
(word,
    synsem:loc:((cat):(head:(verb, Verb),
        subcat:[ (@ np(Ind1)),
            (@ np(Ind3), @ case(acc)),
            (@ np(Ind2), @ case(acc))],
        marking:unmarked),
    cont:(nucleus:(complex_sem,
        struct:(action:(aff_state, afunc:aff_func,
            arg1: Thing1,
            arg2: (Thing3, index:Ind3),

```

```

                                time: Time,
                                manner: no_manner),
                                thematic:ThematicOut),
                                prop:([Thing1, Thing3, Thing2])),
                                quants:[]),
                                conx:backgr:e_set),
    (@ empty_non_loc),
    qstore:e_set)

if (set_select((afunc:effect,
                arg1:(go_event,
                      sem_field:communication,
                      afunc: go_func,
                      arg1: Thing2,
                      arg2:(single_path,
                            afunc:to_path,
                            arg1:(place,
                                  afunc:at_place,
                                  arg1: Thing3))),
                time:Time1)), ThematicIn, Set),
    union((elt:
            (afunc:effect,
              arg1:(have_state,
                    afunc: have_func,
                    arg1: Thing3,
                    arg2: Thing2,
                    time:Time1))),
            elts:e_set), Set, ThematicOut))

morphs
    X becomes X.

% Bring Class lexical rule
%-----
bring_class_to_dative_narrow_lex_rule
(word,
 synsem:loc:((cat):(head:(verb, Verb,
                        vform:bse),
                        subcat:[ (@ np(Ind1)),
                                (@ np(Ind2), @ case(acc)),
                                (@ pp(to, Ind3))]),
              marking:unmarked),
 cont:(nucleus:(complex_sem,
                 struct:(action:(aff_state, afunc:aff_func,
                                arg1: (Thing1, index:Ind1),
                                arg2: (Thing2, index:Ind2),
                                manner: no_manner,
                                time: (Time0, continuous)),
                                thematic:ThematicIn),
                 prop:[Thing1, Thing2, Thing3]),
                 quants:[]),
                 conx:backgr:e_set),
 (@ empty_non_loc),
 qstore:e_set)
**>
(word,
 synsem:loc:((cat):(head:(verb, Verb),
                        subcat:[ (@ np(Ind1)),

```

```

        (@ np(Ind3), @ case(acc)),
        (@ np(Ind2), @ case(acc))],
        marking:unmarked),
    cont:(nucleus:(complex_sem,
        struct:(action:(aff_state, afunc:aff_func,
            arg1: Thing1,
            arg2: (Thing3, index:Ind3),
            manner: no_manner,
            time: (Time0, continuous)),
            thematic:ThematicOut),
        prop:[Thing1, Thing3, Thing2]),
        quants:[]),
        conx:backgr:e_set),
    (@ empty_non_loc),
    qstore:e_set)

if (set_select((afunc:effect,
    arg1:(go_event,
    afunc: go_func,
    arg1: Thing2,
    arg2:(with_to_path,
        arg1:(place,
            afunc:at_place,
            arg1: Thing1),
        arg_2:(place,
            afunc:at_place,
            arg1:(Thing3, deictic))),
    time: (Time1, continuous))), ThematicIn, Set),
union((elt:
    (afunc:effect,
    arg1:(have_state,
    afunc: have_func,
    arg1: Thing3,
    arg2: Thing2,
    time: Time1)),
    elts:e_set), Set, ThematicOut))

morphs
    X becomes X.

% Fulfilling Class "with" preposition lexical rule
%-----
fulfilling_with_prep lex_rule
(word,
    synsem:loc:((cat):(head:(verb, Verb,
        vform:bse),
        subcat:[ (@ np(Ind1)),
            (@ np(Ind3), @ case(acc)),
            (@ pp(with, Ind2))],
        marking:unmarked),
    cont:(nucleus:(complex_sem,
        struct:(action:(aff_state, afunc:aff_func,
            arg1: (Thing1, index:Ind1),
            arg2: (Thing3, index:Ind3),
            time: Time),
            thematic:Thematic),
        prop:[Thing1, Thing3, Thing2]),
        quants:[]),

```

```

        conx:backgr:e_set),
    (@ empty_non_loc),
    qstore:e_set)
**>
(word,
  synsem:loc:((cat):(head:Verb,
    subcat:[ (@ np(Ind1)),
              (@ np(Ind2), @ case(acc)),
              (@ pp(to, Ind3))],
    marking:unmarked),
  cont:(nucleus:(complex_sem,
    struct:(action:(aff_state, affunc:aff_func,
      arg1: Thing1,
      arg2: (Thing2, index:Ind2),
      time: Time,
      manner: no_manner),
    thematic:Thematic),
    prop:[Thing1, Thing2, Thing3]),
    quants:[]),
  conx:backgr:e_set),
  (@ empty_non_loc),
  qstore:e_set)

if set_member((afunc:effect,
  arg1:(go_event,
    afunc: go_func,
    arg1: Thing2,
    arg2:(single_path,
      afunc:to_path,
      arg1:(place,
        afunc:at_place,
        arg1: Thing3)),
    sem_field:possession,
    time: Time)), Thematic)

morphs
  X becomes X.

% Fulfilling Class lexical rule
%-----
fulfilling_to_dative lex_rule
(word,
  synsem:loc:((cat):(head:(verb, Verb,
    vform:bse),
    subcat:[ (@ np(Ind1)),
              (@ np(Ind2), @ case(acc)),
              (@ pp(to, Ind3))],
    marking:unmarked),
  cont:(nucleus:(complex_sem,
    struct:(action:(aff_state, affunc:aff_func,
      arg1: Thing1,
      arg2: (Thing2, index:Ind2, restr:Restr2),
      time: Time),
    thematic:ThematicIn),
    prop:[Thing1, Thing2, Thing3]),
    quants:[]),
  conx:backgr:e_set),

```

```

(@ empty_non_loc),
qstore:e_set)
**>
(word,
synsem:loc:((cat):(head: Verb,
                subcat:[ (@ np(Ind1)),
                        (@ np(Ind3), @ case(acc)),
                        (@ np(Ind2), @ case(acc))],
                marking:unmarked),
cont:(nucleus:(complex_sem,
                struct:(action:(aff_state, afunc:aff_func,
                        arg1: Thing1,
                        arg2: (Thing3, index:Ind3),
                        manner: no_manner,
                        time: Time),
                        thematic:ThematicOut),
                prop:[Thing1, Thing3, Thing2]),
                quants:[]),
                conx:backgr:e_set),
(@ empty_non_loc),
qstore:e_set)

if (poss_prop(Restr2),
    set_select((afunc:effect,
                arg1:(go_event,
                    afunc: go_func,
                    arg1: Thing2,
                    arg2:(single_path,
                        afunc:to_path,
                        arg1:(place,
                            afunc:at_place,
                            arg1: Thing3)),
                    sem_field:possession,
                    time: Time1)), ThematicIn, Set),
    union((elt:
        (afunc:effect,
        arg1:(have_state,
            afunc: have_func,
            arg1: Thing3,
            arg2: Thing2,
            time: Time1)),
        elts:e_set), Set, ThematicOut))

morphs
X becomes X.

poss_prop((elt:(nucleus:(complex_prop,state:action:sem_field:possessional))))
if true.

% "for"-datives
% =====
for_dative lex_rule
(word,
synsem:loc:((cat):(head:(verb, Verb,
                        vform: bse),
                subcat:[ (@ np(Ind1)),
                        (@ np(Ind2), @ case(acc)),
                        (@ pp(for_benefactive, Ind3))],

```

```

        marking:unmarked),
    cont:(nucleus:(complex_sem,
        struct:(action:(aff_state, afunc:aff_func,
            arg1: (Thing1, index:Ind1),
            arg2: (Thing2, index:Ind2),
            time: Time,
            manner: Manner),
        thematic:ThematicIn),
        prop:[Thing1, Thing2, Thing3]),
        quants:[]),
        conx:backgr:e_set),
    (@ empty_non_loc),
    qstore:e_set)
**>
(word,
    synsem:loc:((cat):(head:(verb, Verb),
        subcat:[ (@ np(Ind1)),
            (@ np(Ind3), @ case(acc)),
            (@ np(Ind2), @ case(acc))],
        marking:unmarked),
    cont:(nucleus:(complex_sem,
        struct:(action:(aff_state, afunc:aff_func,
            arg1: Thing1,
            arg2: (Thing3, index:Ind3),
            time: Time,
            manner: Manner),
        thematic:ThematicOut),
        prop:[Thing1, Thing3, Thing2]),
        quants:[]),
        conx:backgr:e_set),
    (@ empty_non_loc),
    qstore:e_set)

if change_thematic(ThematicIn, ThematicOut, Thing1, Thing2, Thing3)
morphs
    X becomes X.

change_thematic(ThematicIn, ThematicIn, _Thing1, Thing2, Thing3) if
set_select((afunc:effect,
    arg1:(go_event,
        afunc: go_func,
        arg1: Thing2,
        arg2: single_path)), ThematicIn, Rest),
set_select((afunc:for_to,
    arg1:(have_state,
        afunc: have_func,
        arg1: Thing3,
        arg2: Thing2)), Rest, Set),
set_member((afunc: means), (ne_set, Set)), !.

change_thematic(ThematicIn, ThematicOut, Thing1, Thing2, Thing3) if
set_select((Effect,
    afunc:effect,
    arg1:(go_event,
        afunc: go_func,
        arg1: Thing2,
        arg2: single_path,
```

```

        time: Time1)), ThematicIn, Rest),
set_select((For_to,
    afunc:for_to,
    arg1:(have_state,
        afunc: have_func,
        arg1: Thing3,
        arg2: Thing2,
        time: time)), Rest, Set),
union((elt:Effect,
    elts:(elt:For_to,
        elts:(elt:
            (afunc:means,
                arg1:(aff_state,
                    afunc: aff_func,
                    arg1: Thing1,
                    arg2: Thing2,
                    time: Time1,
                    manner: no_manner)),
            elts:e_set))), Set, ThematicOut).

% Adjunct Prepositional Phrase lexical rule
% -----
adjunct_prep lex_rule
(word,
synsem:loc:((cat):(head:(prep,
    pform:(adjunctive, Pform),
    mod:none),
    subcat:Subcat),
    cont:Cont,
    conx:Conx),
(@ empty_inher),
qstore:e_set)
**>
(word,
synsem:loc:((cat):(head:(prep,
    pform:Pform,
    mod: @ vp),
    subcat:Subcat),
    cont:Cont,
    conx:Conx),
(@ empty_inher),
qstore:e_set)
morphs
    X becomes X.

% Finite Verb Formation
% -----
% regulars: lexical rule
pres_3s lex_rule
(word,
    synsem:(loc:((cat):(head:(verb,
        vform:bse,
        aux:minus,
        inv:Inv,
        prd:Prd,
```

```

        mod:Mod),
        subcat:[Sub|SubRest],
        marking:Marking),
        cont:Cont,
        conx:Conx),
        non_loc:NL),
qstore:QStore,
qretr:QRetr)
**>
(word,
synsem:(loc:((cat):(head:(verb,
                        vform:fin,
                        aux:minus,
                        inv:Inv,
                        prd:Prd,
                        mod:Mod),
                        subcat:[NewSub|SubRest],
                        marking:Marking),
                        cont:Cont,
                        conx:Conx),
                        non_loc:NL),
qstore:QStore,
qretr:QRetr)

if pres_3s_act(Sub,NewSub)
morphs
  [p,a,y] becomes [p,a,y,s],
  [b,u,y] becomes [b,u,y,s],
  (X,y) becomes (X,i,e,s),    % hurry, carry, fly
  (X,s,s) becomes (X,s,s,e,s), % pass, toss
  X becomes (X,s).

pres_3s_act((NP,@ np(_)),(NP,@ np((per:third,num:sing)),@ case(nom))) if
!,true.
pres_3s_act(X,X) if
true.

pres_non3s lex_rule
(word,
synsem:(loc:((cat):(head:(verb,
                        vform:bse,
                        aux:minus,
                        inv:Inv,
                        prd:Prd,
                        mod:Mod),
                        subcat:[Sub|SubRest],
                        marking:Marking),
                        cont:Cont,
                        conx:Conx),
                        non_loc:NL),
qstore:QStore,
qretr:QRetr)
**>
(word,
synsem:(loc:((cat):(head:(verb,
                        vform:fin,
                        aux:minus,
```

```

        inv:Inv,
        prd:Prd,
        mod:Mod),
        subcat:[NewSub|SubRest],
        marking:Marking),
        cont:Cont,
        conx:Conx),
        non_loc:NL),
    qstore:QStore,
    qretr:QRetr)

    if pres_non3s_act(Sub,NewSub)
    morphs
        X becomes X.

pres_non3s_act((NP,@ np(_)),(NP,@ np((per:(first;second);num:plur)),
        @ case(nom))) if
    true.

% Passive Formation
% -----

passive lex_rule
(word,
    synsem:(loc:((cat):(head:(verb,
        vform:bse,
        aux:minus,
        inv:Inv,
        prd:Prd,
        mod:Mod),
        subcat:[_,
            (loc:((cat):(head:(noun,
                prd:SubPrd,
                mod:SubMod), % ignore case
                subcat:[],
                marking:SubMarking),
                cont:SubCont,
                conx:SubConx),
                non_loc:SubNL)|SubRest]),
            marking:Marking),
        cont:Cont,
        conx:Conx),
        non_loc:NL),
    qstore:QStore,
    qretr:QRetr)
**>
(word,
    synsem:(loc:((cat):(head:(verb,
        vform:pas,
        aux:minus,
        inv:Inv,
        prd:Prd,
        mod:Mod),
        subcat:[(loc:((cat):(head:(noun,
            prd:SubPrd, % no case since
            mod:SubMod), % not fin form

```

```

                                subcat: [],
                                marking: SubMarking),
                                cont: SubCont,
                                conx: SubConx),
                                non_loc: SubNL) | SubRest],
                                marking: Marking),
                                cont: Cont,
                                conx: Conx),
                                non_loc: NL),
qstore: QStore,
qretr: QRetr)

morphs
    give becomes given,
    see becomes seen,
    (X,y) becomes (X,ied),
    (X,e) becomes (X,ed),
    X becomes (X,ed).

% It-Extrapolation
% -----

% regulars: lexical rule
it_extrapolation lex_rule
(word,
    synsem: (loc: ((cat): (head: (Head, vform: bse),
                                subcat: Sub,
                                marking: Mark),
                                cont: Cont,
                                conx: Conx),
                                non_loc: NL),
    qstore: QStore,
    qretr: QRetr)
**>
(word,
    synsem: (loc: ((cat): (head: Head,
                                subcat: ExpSub,
                                marking: Mark),
                                cont: Cont,
                                conx: Conx),
                                non_loc: NL),
    qstore: QStore,
    qretr: QRetr)

if (append(Prev, [(S, @ s(_), loc: (cat): marking: comp) | Rest], Sub),
    append(Rest, [S], NewRest),
    append(Prev, [(@ np(it)) | NewRest], ExpSub))
morphs
    X becomes X.

% Subject Extraction
% -----
subject_extraction lex_rule
(word,
    synsem: (loc: ((cat): (head: (Head, vform: bse), % should look for reltvzr's

```

```

        subcat:Sub,           % also, in order to generate
        marking:Mark),       % SELR version of wh-relativizer
    cont:Cont,                % but that also relies on
    conx:Conx),               % Raising Principle, which
    non_loc:(inherited:(slash:OldSlash, % cannot be implemented yet.
        rel:Rel,
        que:Que),
    to_bind:ToBind)),
qstore:QStore,
qretr:QRetr)
**>
(word,
synsem:(loc:((cat):(head:Head,
    subcat:SESub,
    marking:Mark),
    cont:Cont,
    conx:Conx),
    non_loc:(inherited:(slash:(elt:SlashLoc,elts:OldSlash),
        rel:Rel,
        que:Que),
    to_bind:ToBind)),
qstore:QStore,
qretr:QRetr)

if (append([Prev|Prevs],[(@ s(SCont),loc:(cat):(head:SHead,
    marking:unmarked),
    non_loc:(inherited:(rel:SRel,
        que:SQue),
    to_bind:SToBind))|Rest],Sub),
append([Prev|Prevs],[(@ vp(SCont),loc:(cat):(head:SHead,
    subcat:[(loc:SlashLoc)],
    marking:unmarked),
    non_loc:(inherited:(slash:e_set,
        rel:SRel,
        que:SQue),
    to_bind:SToBind))|Rest],SESub))

morphs
X becomes X.

```

D.3 Grammar Rules

```

% Grammar Rules
% =====

% A sentence is a projection of a finite verb with an empty subcat list, and an
% empty quantifier storage.
%
% NOT STANDARD in HPSG. Added to prevent (main clause) sentences from
% containing quantifiers which have not been retrieved, and non-finite
% verb forms.
%
sentence rule
(sentence,synsem:Synsem,
      qstore:QStore,
      qretr:QRetr)

==>
cat> (phrase,synsem:(Synsem,
      loc:(cat):(head:(verb,
                        vform:fin),
                        subcat:e_list)),
      qstore:(QStore, e_set),
      qretr:QRetr).

schema1 rule
(Mother,phrase,synsem:loc:(cat):subcat:[])
==>
cat> (SubjDtr,non_word,synsem:SubjSyn), % n.b. only one complement permitted
cat> (HeadDtr,phrase),
goal> (head_feature_principle(Mother,HeadDtr),
      inv_minus_principle(Mother),
      subcat_principle(Mother,HeadDtr,[SubjSyn]),
      sre_principle(HeadDtr,[SubjSyn]), % NOT STANDARD in HPSG.
      marking_principle(Mother,HeadDtr),
      spec_principle(SubjDtr,HeadDtr),
      semantics_principle(Mother,HeadDtr,[SubjDtr]),
%      universal_trace_principle: obviated here by parochial
      parochial_trace_principle(SubjDtr),
%      subject_condition: not necessary - sch2,3,or word_promotion_1 did
      nonlocal_feature_principle(Mother,HeadDtr,[SubjDtr]),
      single_rel_constraint(Mother),
      clausal_rel_prohibition(Mother),
      relative_uniqueness_principle(Mother,[SubjDtr,HeadDtr]),
      conx_consistency_principle(Mother,[SubjDtr,HeadDtr]),
      deictic_cindices_principle(Mother,[SubjDtr,HeadDtr])).

schema2 rule
(Mother,phrase,synsem:loc:(cat):subcat:[SubjSyn])
==>
cat> (HeadDtr,word,synsem:loc:(cat):subcat:[SubjSyn|CompSyms]),
goal> synsems_to_non_words(CompSyms,Comps),
cats> (Comps,hd:FirstComp),
goal> (head_feature_principle(Mother,HeadDtr),
      inv_minus_principle(Mother),
      subcat_principle(Mother,HeadDtr,CompSyms),
      sre_principle(HeadDtr,CompSyms), % NOT STANDARD in HPSG.

```

```

marking_principle(Mother,HeadDtr),
spec_principle(FirstComp,HeadDtr),
semantics_principle(Mother,HeadDtr,Comps),
universal_trace_principle(Comps,HeadDtr),
%
parochial_trace_principle: subject not bound yet
subject_condition(CompSyms,SubjSyn),
nonlocal_feature_principle(Mother,HeadDtr,Comps),
single_rel_constraint(Mother),
%
clausal_rel_prohibition: not necessary - mother has non-empty subcat
relative_uniqueness_principle(Mother,[HeadDtr|Comps]),
conx_consistency_principle(Mother,[HeadDtr|Comps]),
deictic_cindices_principle(Mother,[HeadDtr|Comps])).

schema3 rule
(Mother,phrase,synsem:loc:(cat):subcat:[])
==>
cat> (HeadDtr,word,synsem:(loc:(cat):subcat:(SCompSyms,
                                     [SubjSyn|CompSyms])),
      non_loc:to_bind:slash:e_set)),
goal> synsems_to_non_words(SCompSyms,SComps),
cats> (SComps,[Subj|Comps]),
goal> (head_feature_principle(Mother,HeadDtr),
      inv_plus_principle(Mother),
      subcat_principle(Mother,HeadDtr,SCompSyms),
      sre_principle(HeadDtr,SCompSyms), % NOT STANDARD in HPSG.
      marking_principle(Mother,HeadDtr),
      spec_principle(Subj,HeadDtr),
      semantics_principle(Mother,HeadDtr,SComps),
      universal_trace_principle(Comps,HeadDtr), % UTP on FirstComp
      parochial_trace_principle(Subj), % obviated by parochial
      subject_condition(CompSyms,SubjSyn),
      nonlocal_feature_principle(Mother,HeadDtr,SComps),
      single_rel_constraint(Mother),
      clausal_rel_prohibition(Mother),
      relative_uniqueness_principle(Mother,[HeadDtr|SComps]),
      conx_consistency_principle(Mother,[HeadDtr|SComps]),
      deictic_cindices_principle(Mother,[HeadDtr|SComps])).

schema4 rule
(Mother,phrase)
==>
cat> (MarkDtr,phrase,synsem:loc:(cat):(head:mark,
                                     subcat:[])),
cat> (HeadDtr,phrase,synsem:non_loc:to_bind:slash:e_set),
goal> (head_feature_principle(Mother,HeadDtr),
      inv_minus_principle(Mother),
      subcat_principle(Mother,HeadDtr,[]), % no comp-dtrs
      marking_principle(Mother,MarkDtr),
      spec_principle(MarkDtr,HeadDtr),
      semantics_principle(Mother,HeadDtr,[MarkDtr]),
%
% universal_trace_principle: not necessary - no comp-dtrs
% parochial_trace_principle: not necessary - no comp-dtrs
% subject_condition: not necessary - sch2,3 or word_promotion_1 will
nonlocal_feature_principle(Mother,HeadDtr,[MarkDtr]),
single_rel_constraint(Mother),
clausal_rel_prohibition(Mother),
relative_uniqueness_principle(Mother,[MarkDtr,HeadDtr]),

```

```

conx_consistency_principle(Mother, [MarkDtr, HeadDtr]),
deictic_cindices_principle(Mother, [MarkDtr, HeadDtr])).

schema5a rule
(Mother, phrase)
==>
cat> (AdjnDtr, phrase, synsem: loc: (cat): (head: mod: Mod,
                                         subcat: [])),
cat> (HeadDtr, phrase, synsem: (Mod,
                               non_loc: to_bind: slash: e_set)),
goal> (head_feature_principle(Mother, HeadDtr),
      subcat_principle(Mother, HeadDtr, []), % no comp-dtrs
      marking_principle(Mother, HeadDtr),
%      spec_principle: not necessary - no comp-dtrs or marker-dtr
      semantics_principle(Mother, AdjnDtr, [HeadDtr]),
%      universal_trace_principle: not necessary - no comp-dtrs
%      parochial_trace_principle: not necessary - no comp-dtrs
%      subject_condition: not necessary - sch2,3 or word_promotion_1 will
      nonlocal_feature_principle(Mother, HeadDtr, [AdjnDtr]),
      single_rel_constraint(Mother),
      clausal_rel_prohibition(Mother),
      relative_uniqueness_principle(Mother, [AdjnDtr, HeadDtr]),
      conx_consistency_principle(Mother, [AdjnDtr, HeadDtr]),
      deictic_cindices_principle(Mother, [AdjnDtr, HeadDtr])).

schema5b rule
(Mother, phrase)
==>
cat> (HeadDtr, phrase, synsem: (Mod,
                               non_loc: to_bind: slash: e_set)),
cat> (AdjnDtr, phrase, synsem: loc: (cat): (head: mod: Mod,
                                         subcat: [])),
goal> (head_feature_principle(Mother, HeadDtr),
      subcat_principle(Mother, HeadDtr, []), % no comp-dtrs
      marking_principle(Mother, HeadDtr),
%      spec_principle: not necessary - no comp-dtrs or marker-dtr
      semantics_principle(Mother, AdjnDtr, [HeadDtr]),
%      universal_trace_principle: not necessary - no comp-dtrs
%      parochial_trace_principle: not necessary - no comp-dtrs
%      subject_condition: not necessary - sch2,3 or word_promotion_1 will
      nonlocal_feature_principle(Mother, HeadDtr, [AdjnDtr]),
      single_rel_constraint(Mother),
      clausal_rel_prohibition(Mother),
      relative_uniqueness_principle(Mother, [AdjnDtr, HeadDtr]),
      conx_consistency_principle(Mother, [AdjnDtr, HeadDtr]),
      deictic_cindices_principle(Mother, [AdjnDtr, HeadDtr])).

schema6 rule
(Mother, phrase)
==>
cat> (FillDtr, phrase, synsem: (loc: FillLoc,
                               non_loc: inherited: slash: e_set)),
cat> (HeadDtr, phrase, synsem: (loc: (cat): (head: (verb,
                                              vform: fin),
                                              subcat: []),
                               non_loc: (inherited: slash: HeadSlashes,
                                           to_bind: slash: (elt: FillLoc,

```

```

                                                    elts:e_set))))),
goal> (set_member(FillLoc,HeadSlashes),
      head_feature_principle(Mother,HeadDtr),
      subcat_principle(Mother,HeadDtr,[]),      % no comp_dtrs
      marking_principle(Mother,HeadDtr),
%      spec_principle: not necessary- no comp-dtrs or marker-dtr
      semantics_principle(Mother,HeadDtr,[FillDtr]),
%      universal_trace_principle: not necessary - no comp-dtrs
%      parochial_trace_principle: not necessary - no comp-dtrs
%      subject_condition: not necessary - sch2,3 or word_promotion_1 will
      nonlocal_feature_principle(Mother,HeadDtr,[FillDtr]),
      single_rel_constraint(Mother),
      clausal_rel_prohibition(Mother),
      relative_uniqueness_principle(Mother,[FillDtr,HeadDtr]),
      conx_consistency_principle(Mother,[FillDtr,HeadDtr]),
      deictic_cindices_principle(Mother,[FillDtr,HeadDtr])).

word_promotion_0 rule
(phrase,synsem:Synsem,
  qstore:QStore,
  qretr:QRetr)
==>
cat> (word,synsem:(Synsem,loc:(cat):subcat:[],
                    non_loc:to_bind:slash:e_set),
      qstore:QStore,
      qretr:QRetr).

word_promotion_1 rule
(phrase,synsem:Synsem,
  qstore:QStore,
  qretr:QRetr)
==>
cat> (word,synsem:(Synsem,loc:(cat):subcat:[SubjSyn],
                    non_loc:to_bind:slash:e_set),
      qstore:QStore,
      qretr:QRetr),
goal> subject_condition([],SubjSyn).      % no other comps

% Macros
% =====

% Added by Karin
%-----
pp(Pform, Ind) macro
  loc:((cat):(head:(prep, pform:Pform),
                subcat:[]),
        cont:index:Ind).

pp(Pform, Ind, Restr) macro
  loc:((cat):(head:(prep, pform:Pform),
                subcat:[]),
        cont:(index:Ind,
              restr:Restr)).

prep(Pform) macro
  word,
```

```

synsem:loc:((cat):(head:(prep,
                        pform:Pform,
                        mod:none),
                        subcat:[(@ np(Ind, Restr), @ case(acc))]),
            cont:(index:(Ind,ref),
                  restr:Restr),
            conx:backgr:e_set),
(@ empty_inher),
qstore:e_set.

thing(Ind) macro
  thing,
  index:Ind,
  restr:e_set.

thing_spec(Ind, Thing) macro
  thing,
  index:(ref,Ind,
          per:third,
          num:sing,
          gen:neut),
  restr:(elt:
        (nucleus:(Thing, inst: Ind),
         quants:[]),
         elts:e_set).

thing(Ind, Restr) macro
  thing,
  index:Ind,
  restr:Restr.

vp macro
  loc:((cat):(head:verb,
              subcat:[synsem])).

np(Ind, Restr) macro
  loc:((cat):(head:noun,
              subcat:[]),
        cont:(index:Ind,
              restr:Restr)).

%-----
% (from original Penn and Carpenter grammar)

np(Ind) macro % p. 16
  loc:((cat):(head:noun,
              subcat:[]),
        cont:index:Ind). % this one is NP "sub" i in the book, not NP:i

nbar(Cont) macro % p. 46
  loc:((cat):(head:noun,
              subcat:[(@ detp)]),
        cont:Cont).

case(Case) macro
  loc:(cat):head:case:Case.

```

```

s(Proposition) macro
  loc:((cat):(head:verb,
             subcat:[]),
    cont:Proposition).

vp(Proposition) macro
  loc:((cat):(head:verb,
             subcat:[synsem]),
    cont:Proposition).

detp macro % p. 45
  loc:(cat):(head:det,
             subcat:[]).

empty_non_loc macro
  synsem:non_loc:(inherited:(que:e_set,
                             rel:e_set,
                             slash:e_set),
    to_bind:(que:e_set,
             rel:e_set,
             slash:e_set)).

empty_inher macro
  synsem:non_loc:inherited:(que:e_set,
                             rel:e_set,
                             slash:e_set).

% Principles
% =====
% selection_restriction_enforcement_principle(HeadDtr, Comp-Dtr-Synsems)
% NOT STANDARD in HPSG.
%-----

%% sre_principle
% This is the principle which governs verb selection restriction
% enforcement.
sre_principle((synsem:(loc:((cat):(head:verb,
                                subcat:Subcat),
                          cont:nucleus:(top_level_desc,
                                prop:Proplist)))))
    Comps) if
  !, sre_principle_match(Subcat, Proplist, Comps).

sre_principle(_,_) if true.

%% sre_principle_match
% A helper function for sre_principle which ensures that the parts of the
% property list which are relevant to the current complements are
% picked out.
sre_principle_match(Subcat, Proplist, Subcat) if
  sre_principle_help(Subcat, Proplist).
sre_principle_match([_H|Comps], [_HP|TP], Comps) if
  sre_principle_help(Comps, TP).

```

```

%% sre_principle_help
% A helper function for sre_principle which ensures that the
% restrictions specified in the property list (ReqRestr) for each
% complement noun or preposition projection are a subset of the
% properties of the actual noun or preposition projections which
% fills the corresponding argument position.
sre_principle_help([],_) if
    true.
sre_principle_help([(loc:((cat):(head:(prep;noun),
                                subcat:[]),
                                cont:(index:Ind,
                                        restr:Restr)))|RestSubcat],
                    [(index:Ind,restr:ReqRestr)|RestProp]) if
    subset(ReqRestr, Restr),
    sre_principle_help(RestSubcat, RestProp).
sre_principle_help([(loc:((cat):(head:(func;adj;verb;reltvzr))))|RestSubcat],
                    Proplist) if
    sre_principle_help(RestSubcat, Proplist).

% head_feature_principle(Mother,Head-Daughter)
%-----
head_feature_principle(synsem:loc:(cat):head:X,synsem:loc:(cat):head:X) if
    true.

% subcat_principle(Mother,Head-Daughter,Comp-Dtr-Synsems)
%-----
subcat_principle((synsem:loc:(cat):subcat:MSub),(synsem:loc:(cat):subcat:HSub),
                  CompSyns) if
    append(MSub,CompSyns,HSub).

% marking_principle(Mother,Mark-Dtr)
%-----
% Mark-Dtr is marker-dtr, if any, o.w. head-dtr
marking_principle(synsem:loc:(cat):marking:Mark,
                  synsem:loc:(cat):marking:Mark) if
    true.

% spec_principle(Spec-Dtr,Head-Dtr)
%-----
% Spec-Dtr is either mark-dtr or first comp-dtr
spec_principle((synsem:loc:(cat):head:Head),synsem:HeadSynsem) if
    specp_act(Head,HeadSynsem).

specp_act(subst,_) if      % substantive head
    true.
specp_act(spec:X,X) if    % functional head
    true.

% semantics_principle(Mother,Semantic-Head,Other-Dtrs)
%-----
% Semantic-Head is adjunct-dtr, if any, o.w. head-dtr
semantics_principle((qstore:MQStore,
                     qretr:MRetr,
                     synsem:loc:cont:MCont),
                    (SHead,synsem:loc:cont:SCont),ODtrs) if

```

```

qstores_of([SHead|ODtrs],e_set,DQStore),
semp_act(SCont,MRetr,MQStore,MCont,DQStore).

semp_act((psoa,nucleus:Nucl,
          quants:SQuants),MRetr,MQStore,(nucleus:Nucl,
          quants:MQuants),DQStore) if
    !,set_sublist(MRetr,DQStore,MQStore),    % part (a)
    append(MRetr,SQuants,MQuants).           % part (b)
semp_act(Cont,[],QStore,Cont,QStore) if      % parts (a) and (b)
    true.

% universal_trace_principle(Comp-Dtrs,Head-Dtr)
%-----
% The situation of the trace sort in the subsumption hierarchy, and the
% type constraints on the participants of schemata guarantee that traces
% will only appear as subcategorized elements. The following ensures
% that they will only appear as subcategorized by substantives.
universal_trace_principle([(trace)|_],HeadDtr) if
    !,utp_act(HeadDtr).
universal_trace_principle([_|Comps],HeadDtr) if
    universal_trace_principle(Comps,HeadDtr).
universal_trace_principle([],_) if
    true.

utp_act((synsem:loc:(cat):head:subst)) if    % act predicate necessary for
    true.                                     % proper placement of cut above

% parochial_trace_principle(First-Comp-Dtr)
%-----
% strict subcategorization: excludes that-trace sentences
parochial_trace_principle(trace) if
    !,fail.
parochial_trace_principle(_) if
    true.

% subject_condition(Other-Comp-Dtr-Synsems,Subj-Dtr-Synsem)
%-----
subject_condition([],non_loc:inherited:slash:e_set) if
    true.
subject_condition([(non_loc:inherited:slash:ne_set)|_],_) if
    !,true.
subject_condition([(non_loc:inherited:slash:e_set)|CompSynRest],SubjSyn) if
    subject_condition(CompSynRest,SubjSyn).

% nonlocal_feature_principle(Mother,Head-Dtr,Other-Dtrs)
%-----
nonlocal_feature_principle((synsem:non_loc:inherited:(slash:MISlash,
          que:MIQue,
          rel:MIRel)),
          (HeadDtr,synsem:non_loc:to_bind:(slash:HTSlash,
          que:HTQue,
          rel:HTRel)),
          ODtrs) if
    islashes_of([HeadDtr|ODtrs],e_set,DISlash),
    iques_of([HeadDtr|ODtrs],e_set,DIQue),
    irels_of([HeadDtr|ODtrs],e_set,DIRel),
    set_diff(HTSlash,DISlash,MISlash),

```

```

set_diff(HTQue,DIQue,MIQue),
set_diff(HTRel,DIRel,MIRel).

% relative_uniqueness_principle(Mother,Dtrs)
%-----
% parochial(certain dialects): constrains the result of non-local feature
% principle to prevent parasitic relatives
relative_uniqueness_principle(synsem:non_loc:inherited:rel:Rel,Dtrs) if
    rup_act(Rel,Dtrs).

rup_act(e_set,_) if
    true.
rup_act((elt:X,elts:Xs),Dtrs) if
    rup_elt(Dtrs,X),
    rup_act(Xs,Dtrs).

rup_elt([],_) if
    true.
rup_elt([(synsem:non_loc:inherited:rel:DRel)|DtrsRest],X) if
    set_member_eq(X,DRel),
    !,rup_elt_act(DtrsRest,X).    % belongs to one daughter
rup_elt([_|DtrsRest],X) if
    rup_elt(DtrsRest,X).

rup_elt_act([],_) if
    true.
rup_elt_act([(synsem:non_loc:inherited:rel:DRel)|DtrsRest],X) if
    (\+ set_member_eq(X,DRel)),
    rup_elt_act(DtrsRest,X).    % but no more than one

% conx_consistency_principle(Mother,Dtrs)
%-----
conx_consistency_principle((synsem:loc:conx:backgr:MBackgr),
    Dtrs) if
    backgrs_of(Dtrs,e_set,MBackgr).

% deictic_cindices_principle(Mother,Dtrs)
%-----
deictic_cindices_principle((synsem:loc:conx:c_inds:MCinds),
    Dtrs) if
    dcip_act(Dtrs,MCinds).

dcip_act([],_) if
    true.
dcip_act([(synsem:loc:conx:c_inds:DCinds)|DRest],DCinds) if
    dcip_act(DRest,DCinds).

% inv_minus_principle(Mother)
%-----
% parochial: if inv exists, it must be minus
inv_minus_principle(synsem:loc:(cat):head:inv:Inv) if
    !,imp_act(Inv).    % inv is approp. and minus
inv_minus_principle(_) if
    true.    % or inapprop.

```

```

imp_act(minus) if
    true.

% inv_plus_principle(Mother)
%-----
% parochial: if inv exists, it must be plus
inv_plus_principle(synsem:loc:(cat):head:inv:Inv) if
    !,ipp_act(Inv).           % either inv is approp. or causes failure
inv_plus_principle(_) if
    true.                     % and succeeds here

ipp_act(plus) if
    true.

% single_rel_constraint(Sign)
%-----
% parochial: Rel set can't have more than one element
% enforced on words and trace as type constraint; enforced on phrases as
% procedural attachment to rules
single_rel_constraint(synsem:non_loc:inherited:rel:e_set) if
    true.
single_rel_constraint(synsem:non_loc:inherited:rel:elts:e_set) if
    true.

% clausal_rel_prohibition(Sign)
%-----
% parochial: Sentences must have empty Rel set
clausal_rel_prohibition((synsem:non_loc:inherited:rel:e_set)) if % empty Rel
    true.
clausal_rel_prohibition((synsem:(non_loc:inherited:rel:ne_set,
                                loc:(cat):head:(func
                                                ;adj
                                                ;noun
                                                ;prep
                                                ;reltvzr)))) if
    true.                     % not a verb projn.
clausal_rel_prohibition((synsem:(non_loc:inherited:rel:ne_set,
                                loc:(cat):(head:verb,
                                                subcat:ne_list)))) if
    true.                     % not a sentence

% Utilities
% =====
% subset(+S1, +S2) is S1 a subset of S2?
subset(S1, S1) if
    !, true.
subset(e_set, _S2) if
    !, true.
subset((elt:X, elts:Xs), S2) if
    set_member(X, S2), !,
    subset(Xs, S2).

% union(?set1,+set2,?union)
union(e_set,Xs,Xs) if
    true.
union((elt:X,elts:Xs),Ys,Zs) if

```

```

    set_member_eq(X,Ys),
    !,union(Xs,Ys,Zs).
union((elt:X,elts:Xs),Ys,(elt:X,elts:Zs)) if
    set_select(X,Ys,YsRest),
    union(Xs,YsRest,Zs).
union((elt:X,elts:Xs),Ys,(elt:X,elts:Zs)) if
    union(Xs,Ys,Zs).

set_member(X,(elt:X)) if
    true.
set_member(X,(elts:S)) if
    set_member(X,S).

set_member_eq(X,(elt:Y)) if
    (X = $\emptyset$  Y).
set_member_eq(X,(elts:S)) if
    set_member_eq(X,S).

set_select(X,(elt:X,elts:Xs),Xs) if
    true.
set_select(Member,(elt:X,elts:Xs),(elt:X,elts:Rest)) if
    set_select(Member,Xs,Rest).

% set_select_eq(+member,+set,?rest)
set_select_eq(X,(elt:Y,elts:Xs),Xs) if
    (X = $\emptyset$  Y).
set_select_eq(Member,(elt:X,elts:Xs),(elt:X,elts:Rest)) if
    set_select_eq(Member,Xs,Rest).

append([],Xs,Xs) if
    true.
append([H|T1],L2,[H|T2]) if
    append(T1,L2,T2).

% selectors
%-----
% in the X_of predicates, testing first for e_set means that if we don't
% specify that feature, then, by default, it is the empty set

backgrs_of([],MBackgr,MBackgr) if
    true.
backgrs_of([(synsem:loc:conx:backgr:e_set)|DRest],Accum,MBackgr) if
    backgrs_of(DRest,Accum,MBackgr),
    !.
backgrs_of([(synsem:loc:conx:backgr:DBackgr)|DRest],Accum,MBackgr) if
    union(Accum,DBackgr,NewAccum),
    backgrs_of(DRest,NewAccum,MBackgr).

qstores_of([],QStores,QStores) if
    true.
qstores_of([(qstore:e_set)|Dtrs],Accum,QStores) if
    qstores_of(Dtrs,Accum,QStores),
    !.
qstores_of([(qstore:DQStore)|Dtrs],Accum,QStores) if
    union(Accum,DQStore,NewAccum),
    qstores_of(Dtrs,NewAccum,QStores).

```

```

islashes_of([],ISlash,ISlash) if
    true.
islashes_of([(synsem:non_loc:inherited:slash:e_set)|Dtrs],Accum,ISlash) if
    islashes_of(Dtrs,Accum,ISlash),
    !.
islashes_of([(synsem:non_loc:inherited:slash:DISlash)|Dtrs],Accum,ISlash) if
    union(Accum,DISlash,NewAccum),
    islashes_of(Dtrs,NewAccum,ISlash).

iques_of([],IQue,IQue) if
    true.
iques_of([(synsem:non_loc:inherited:que:e_set)|Dtrs],Accum,IQue) if
    iques_of(Dtrs,Accum,IQue),
    !.
iques_of([(synsem:non_loc:inherited:que:DIQue)|Dtrs],Accum,IQue) if
    union(Accum,DIQue,NewAccum),
    iques_of(Dtrs,NewAccum,IQue).

irels_of([],IRel,IRel) if
    true.
irels_of([(synsem:non_loc:inherited:rel:e_set)|Dtrs],Accum,IRel) if
    irels_of(Dtrs,Accum,IRel),
    !.
irels_of([(synsem:non_loc:inherited:rel:DIRel)|Dtrs],Accum,IRel) if
    union(Accum,DIRel,NewAccum),
    irels_of(Dtrs,NewAccum,IRel).

set_sublist([],Set,Set) if
    true.
set_sublist([X|Subs],Set,RestSet) if
    set_select(X,Set,Rest),
    set_sublist(Subs,Rest,RestSet).

set_diff(e_set,Set,Set) if    %first arg should be instantiated, so cut
    !,true.                  % in case it isn't, and it should also be subset
                             % of the second

set_diff((elt:X,elts:Xs),Set,Diff) if
    set_select_eq(X,Set,Rest),
    set_diff(Xs,Rest,Diff).

synsems_to_non_words([],[]) if    % cut is very important - nothing has
    !,true.                        % guaranteed that inputs are sufficiently
                                   % instantiated
synsems_to_non_words([Syn|Synsems],[(non_word,synsem:Syn)|Signs]) if
    synsems_to_non_words(Synsems,Signs).

```

Bibliography

- Carpenter, B. (1993). ALE: The Attribute Logic Engine user's guide. Version Beta.
- Chomsky, N. (1986). *Knowledge of language: Its nature, origin, and use*. Praeger, New York.
- Gruber, J. (1965). *Studies in Lexical Relations*. PhD thesis, MIT, Cambridge, MA.
- Hardie, R. (1990). *Collins Gem English Grammar*. HarperCollins Publishers. Editors: Hanks, Patrick and Grandison, Alice.
- Jackendoff, R. (1983). *Semantics and Cognition*. The MIT Press, Cambridge, Massachusetts.
- Jackendoff, R. (1990). *Semantic Structures*. Current Studies in Linguistics series. The MIT Press, Cambridge, Massachusetts.
- Pinker, S. (1989). *Learnability and cognition: the acquisition of argument structure*. Learning, Development, and Conceptual Change series. The MIT Press, Cambridge, Massachusetts.
- Pollard, C. and Sag, I. (1987). *Information-based syntax and semantics*, volume 1: Fundamentals. Center for the Study of Language and Information, Stanford, CA.
- Pollard, C. and Sag, I. (1994). *Head-Driven Phrase Structure Grammar*. University of Chicago Press.
- Pustejovsky, J. (1991). The generative lexicon. *Computational Linguistics*, 17(4).
- Talmy, L. (1985). Lexicalization patterns: Semantic structure in lexical forms. In Shopen, T., editor, *Language typology and syntactic description*, volume 3: Grammatical categories and the lexicon. Cambridge University Press.
- Zwarts, J. and Verkuyl, H. (1994). An algebra of conceptual structure; An investigation into Jackendoff's Conceptual Semantics. *Linguistics and Philosophy*, 17:1–28.